Problem definition: We formulate a novel class of online matching problems with learning. In these problems, randomly arriving customers must be matched to perishable resources so as to maximize a total expected reward. The matching accounts for variations in rewards among different customer-resource pairings. It also accounts for the perishability of the resources. For concreteness, we focus on healthcare platforms, but our work can be easily extended to other service applications.

Academic / Practical Relevance: Our work belongs to the online resource allocation streams in service system.

Methodology: We propose the first online algorithm for contextual learning and resource allocation with perishable resources, the Two-Phase algorithm which explores and exploits in distinct phases.

Results: We prove that our algorithm achieve an expected regret per period of \(O(\gamma^{-\frac{1}{3}})\), where \(\gamma\) is a scaling that reflects the size of the system. We also prove that no online algorithm can achieve an expected regret per period that is better than \(\Theta(\gamma^{-\frac{1}{2}})\) for this class of problems.

Managerial Implications: We propose a pioneer algorithm that helps service system to optimize resource allocation decisions while learn the uncertain reward of matching customer-resource pairings.

Key words: Analysis of algorithms, Approximations/heuristic, Cost analysis
of the objectives of a platform is to improve the matching of customer with product and service types.

In this paper, we address an online matching problem, where randomly arriving demands are matched to available resources so as to maximize a total expected reward. The matching accounts for variations in rewards among different customer-resource pairings. It also accounts for the possible perishability of the resources. For concreteness, we focus on healthcare platforms, but our work can be easily extended to other service applications.

In healthcare, ZocDoc and HaoDF are examples of web-based platforms that provide convenient healthcare services to patients. These platforms display availability information of physicians, allowing patients to choose appointments with these physicians. They also provide physician recommendations to patients. In recent years, electronic medical records are becoming increasingly popular. These records provide platforms with patient and physician information that can be used to make predictions about the volume of patient demands as well as the quality of patient-physician matches. Using this information, platforms are potentially able to offer unprecedented levels of customized service.

In platforms such as the ones described above, there are several key features that are important from a practical perspective. First, patients (henceforth called “demands”), and physicians (henceforth called “resources”), might be highly varied, and the suitability of matching demand to resources very often depend on the characteristics of both sides. For example, the suitability of matching a patient to a physician depends on both the physician’s expertise, availability and location, as well as the patient’s ailment, schedule, and location. Second, without extensive prior information, it might be very difficult to predict the quality of a match. In these cases, feature information can be very useful in making such predictions based on historical data. Examples of feature information include search histories, electronic health records, past reviews, and disclosures by the physicians, etc. By analyzing feature information, the platform can obtain better estimate the quality of a match. Third, service resources are often perishable, so that after a time, they can no longer be used. Therefore, the system dynamics must depend on the status of available resources. Finally, the arrivals of demands are usually non-stationary. Upon each arrival, a decision should be made to achieve two conflicting objectives: one is to learn an unknown reward distributions which requires exploring new pairings of demands and resources; the
other is to maximize the total reward given current information, which tends to exploit the pairings that yield the highest rewards based on past observations.

We formulate the resulting problem as an online matching problem with learning. We model a network with multiple physicians, each providing a fixed number of service slots on different service days. Patients’ requests for appointment arrive according to non-stationary Poisson processes. Upon each arrival, one slot on a specific service day for one specific provider should be assigned to the patient, or the patient should be rejected. Feature information about the patient will be observed upon arrival. The feature information of both patient and physicians determine the reward of their pairing. In this way, the feature information of the current patient at a given time can be considered as a context for the decision taking place at that time. We assume that the reward as a function of the feature information is not known at the beginning of the horizon, and so must be learned over time. We allow the reward to depend explicitly on the service delay, or the amount of time that a customer must wait before being served.

We propose the first online algorithm for contextual learning and resource allocation with perishable resources. Learning in these environment is challenging for three reasons. First, we face uncertainty in demand arrivals. Unlike in multi-armed bandit problems, we cannot proactively sample the rewards. Rather, we must wait for customers, or contexts, to arrive. Second, we need to strategically allocate finite perishable resources to generate diverse customer-resource pairings, in order to effectively learn the reward as a function of these pairings. Our learning therefore, is constrained by the availability of perishing resources. Finally, because of the limited availability of perishing resources, our policy must choose its actions based on the real-time state of the resources, rather than statically. We prove that algorithm achieves an expected regret per period of $O(\frac{1}{\gamma})$, where $\gamma$ is a scaling that reflects the size of the system. We also prove that no algorithm can achieve an expected regret per period that is better than $\Theta(\gamma^{-\frac{1}{2}})$ for this class of problems. With data from HaoDF, one of the most popular online healthcare platform China, we carry out a case study in a dermatology department that showcases numerical results that are consistent with our theoretical proofs.

2. Literature review

We review briefly three streams of literature that are closest to our work, namely, the literature on contextual bandit, online matching, and advance scheduling.
2.1. Contextual Bandits

Our problem can be cast as a contextual multi-armed-bandit problem. In these problems, the reward of an arm is partly determined by exogenous random contextual information that is revealed at each round. Specifically, our work is most similar to the Contextual Bandits with Knapsack (CBwK) problem, where each pulled arm consumes a certain amount of resources, each with limited capacity. Past works all make assignments using static probabilities. In particular, the benchmark optimal policy in these works is simply a static mapping from customers (contexts) to resources. As such, these algorithms are not appropriate for non-stationary environments, such as those with perishing resources and non-stationary demand arrivals that we study in this paper. In contrast, we make learning and allocation decisions using time-dependent, dynamic policies that account for the real-time availability of resources.

There are many papers that study the contextual bandits problem and its special case, linear contextual bandits, where the rewards are unknown linear functions of the contexts. Auer (2003) proposes the first algorithm called LINREL, which is based on the Upper Confidence Bound (UCB) algorithm. This work motivates the LINUCB algorithm in Li, Wei, Langford, and Schapire (2010). Chu, Li, Reyzin, and Schapire (2011) provides regret analysis for a variant of the LINUCB algorithm. However, all of these papers do not have capacity constraints as we do. Agrawal and Devanur (2014) consider the general problem of Bandits with Concave Reward and Convex Constraints (BwCR), and make an extension to the case with linear contexts. They examine a family of UCB algorithms. Badanidiyuru, Langford, and Slivkins (2014) extend the general contextual bandit problem with arbitrary policy sets to allow budget constraints. They also propose an algorithm that tries to explore as much as possible while avoiding obvious suboptimal decisions. Agrawal, Devanur, and Li (2015) propose an efficient algorithm based on the problem formulation of Badanidiyuru, Langford, and Slivkins (2014), and also consider an extension to concave objectives. Agrawal and Devanur (2016) incorporate knapsack constraints into the linear contextual bandit problem, and use optimistic estimates constructed from confidence ellipsoids to form adjusted estimated rewards, based on which actions are selected. Table 1 summarizes the most closely related works in this literature.

Johari, Kamble, and Kanoria (2017) use the Bandits with Knapsack (BwK) formulation to study a different job assignment problem and analyzes regret bounds similar to
ours. However, they do not address contextual information in their formulation. In the revenue-management literature, there are several works dealing with reoptimization or self-adjusting policy for network revenue management with demand learning, such as Jasin (2014) and Chen and Ross (2014). In these settings, there is no context information. As far as we are aware, Ferreira, Simchilevi, and Wang (2015) is the only work that addresses a contextual learning problem and proposes an adaptive policy as we do. Their algorithm is based on the traditional Thompson sampling method. However, in this dynamic pricing literature, the learning objective is the price-demand relationship, and to that end, the demand is always assumed to be time-homogeneous. Compared to Ferreira, Simchilevi, and Wang (2015), the quantity to be learned in our setting is the reward distribution as it depends on the context and on the demand-resource pairing.

2.2. Online Matching

Our work is also related to the research on online bipartite matching, where resources are assigned to randomly arriving demands in an online manner. Traditionally, the performance of the online algorithm is evaluated by a competitive ratio (CR), i.e., a ratio between the rewards of an online algorithm and an optimal offline algorithm that knows all the demand information beforehand. The main distinction between our work and this literature is that this literature does not address learning, whereas we do.

Many of the works on online matching propose algorithms use a primal-dual paradigm, for example Devanur (2009) for adwords allocation, Feldman, Henzinger, Korula, Mirrokni, and Stein (2010) for stochastic packing, and Agrawal, Wang, and Ye (2009) for online linear programming. These papers model a random order arrival of demand, where the arrival set is generated by an adversary, and then realized in a random order. In the primal-dual paradigm, the online algorithm observes the demands in some early periods and solves a primal linear program based on the observed history. They subsequently use dual prices to make decisions on the rest of the arrivals. Devanur and Jain (2012) and Chen and Wang (2015) extend these problems to include general concave objectives. Hu and Zhou (2016) study an online matching problem where demand and supply units of various types arrive in random quantities in each period, and unmatched demand and supply units are carried over to the next period, with abandonment and waiting or holding costs. Wang, Truong, and Bank (2015) study an online resource allocation problem with non-stationary Poisson arrival processes. The paper designs two algorithms that split the multi-resource problem.
Agrawal and Devanur (2016)  
Linear reward and objective  
Stationary and discrete  
Reward and consumption distributions for given contexts  
No  
Linear knapsack  
$O\left(\frac{\sqrt{\ln(\gamma)}}{\gamma}\right)$  
Horizon length

Agrawal and Devanur (2014)  
Linear reward and concave objective  
Stationary and discrete  
Reward and consumption distributions for given contexts  
No  
Convex constraints  
$O\left(\frac{\ln(\gamma)}{\gamma}\right)$  
Horizon length

Agrawal, Devanur, and Li (2015)  
General reward and concave objective  
Stationary and discrete  
Reward and consumption distributions for given contexts  
No  
Linear knapsack  
$O\left(\frac{\ln(\gamma)}{\gamma}\right)$  
Horizon length

Badanidiyuru, Langford, and Slivkins (2014)  
General reward and objective  
Stationary and discrete  
Reward and consumption distributions for given contexts  
No  
Linear knapsack  
$O\left(\frac{\ln(\gamma)}{\gamma}\right)$  
Horizon length

General reward and linear objective  
Stationary and discrete  
Price-dependent demand distribution  
Yes  
Linear knapsack  
$O\left(\frac{1}{\sqrt{\gamma}}\right)$  
Horizon length and capacity

Our work  
Both linear  
Non-stationary and continuous  
Reward distributions for given contexts  
Yes  
Linear knapsack  
$O\left(\gamma^{-\frac{1}{2}}\right)$  
Capacity and arrival rates

Table 1  
Comparison with relevant literature on resource allocation with learning. The scaling parameter $\gamma$ is different in each paper.

into multiple single-resource problems. They derive a constant competitive ratio of $1/2$ for their algorithms, and show that this is the best achievable constant bound. We use one of their algorithms in our paper as a subroutine. However, rather than analyzing the algorithm’s competitive ratio, we analyze its regret, which is substantially different. The former can be thought of as a multiplicative difference, whereas the latter, an additive difference. In particular, the worst-case instance for the competitive ratio is distinct from its counterpart for the regret.

2.3. Advance Scheduling

From the perspective of application, our model can be cast as an advance scheduling problem. Advance scheduling refers to the problem of inter-slot allocation of patients to
healthcare resources. However, previous literature has assumed that the reward parameters for such assignments are known. In practice, the reward is highly variable and cannot be easily quantified. We contribute to this literature by treating the reward as an unknown quantity that must be learned.

In the advance scheduling literature, Patrick, Puterman, and Queyranne (2008) consider the allocation of service slots to patients with different priorities with delay and overtime costs. The paper shows that this advance scheduling problem suffers from the “curse of dimensionality” and solves the problem using dynamic programming approximation. Liu, Ziya, and Kulkarni (2010) focus on the no-show and cancellation behaviors in advance scheduling, and show that tracking the wait times of patients easily makes the problem intractable. Thus, they proposes heuristic policies for the problem. Gupta and Wang (2008) and Feldman, Liu, Topaloglu, and Ziya (2014) incorporate patient preferences over slots or service slots using explicit choice models such as the multinomial logit (MNL) model. Liu, van de Ven, and Zhang (2016) also studies the appointment scheduling problem with heterogeneous patient preferences, and develops two models, a non-sequential scheduling model and a sequential scheduling model, to capture different types of interactions. Truong (2015) is the first to come up with analytical results for the advance scheduling problem with two patient classes, an urgent demand class and a regular demand class, by assuming that patients take the earliest appointment offered and do not differentiate among providers.

3. Problem formulation
3.1. Model
Consider a continuous planning horizon of length $T$. The horizon can be divided into cycles of length $L$, and there are totally $K = \frac{T}{L}$ cycles. Time 0 corresponds to the beginning of the first cycle, whereas time $T$ corresponds to the end of last cycle.

There are $J$ types of resources indexed by $j = 1, 2, \ldots, J$. Each resource has a capacity of $c_j$. All resources are perishable but recurrent in the sense that in each cycle, a new resource of type $j$ with the same capacity $c_j$ becomes available and the previous resource of type $j$ expires. Accordingly, let $s_{jk}$ be the expiration time, or service time for resource type $j$ in the $k$-th service cycle. Each resource $j$ that is offered at the $k$-th service slot, called $(j,k)$, can be described by a vector of features $[s_{jk}; y_j] \in \mathcal{Y}^{N+1}$. We assume all resources are known at the beginning of the horizon.
There are $I$ types of customers indexed by $i = 1, 2, ..., I$. Customers of type $i$ arrive according to a non-stationary Poisson process with rate $\lambda_i(t)$, for $0 \leq t \leq T$, $i = 1, 2, ..., I$. We assume that the arrival rates $\lambda_i(t)$, $i = 1, 2, ..., I$, are known. Each customer $i$ arriving at time $t$, called $(t, i)$, is described by a vector of features $[t; x_i] \in \mathcal{X}^{M+1}$. Recall that the customer features constitute the context at time $t$.

A customer must be matched to a resource within $W$ time units of his or her arrival. We call $W$ the booking window. When a resource $(j, k)$ is matched to a customer $(t, i)$, $t \leq s_{jk} \leq t + W$, a reward of $r_{ijk}(t) = x_i^T Ay_j + a(s - t) + \epsilon_t$ is generated, where $A$ is an unknown $M \times N$ matrix, $a$ is an unknown scalar that measures the effect of the service delay $s - t$, and $\epsilon$ is a zero-mean random variable with known variance $\sigma$. We assume that $\epsilon$ is a Normal random variable $N(0, \sigma^2)$. Without loss of generality, we assume $E[r(\cdot)] \leq 1$, and for notational convenience we use $r(\cdot) \leq 1$ in this paper to denote $E[r(\cdot)] \leq 1$.

The resource, once assigned, will have its capacity reduced by one unit. We let $K_W = \frac{W}{T} \in \mathbb{Z}^+$ denote the total number of cycles within a booking window. In other words, for each resource type $j$, there are $K_W$ feasible ways in which a customer can be assigned to a type-$j$ resource.

3.2. Regret

We benchmark an algorithm by comparing it against an optimal dynamic policy that knows the reward function $r(\cdot)$, i.e. both $A$ and $a$. Due to the intractability of computing an optimal dynamic policy, we use the optimal value $V^{OPT}$ of a static offline policy $OPT$ instead. It is well known that $V^{OPT}$ is an upper bound on the expected reward of an optimal dynamic policy.

To this end, let $p^*_{ijk}(t)$ be an optimal solution of the following LP:

$$\max_{p_{ijk}(t), k \in \{1, 2, ..., K\}} \sum_{i=1}^{I} \sum_{j=1}^{J} \sum_{k=1}^{K} \int_{0}^{T} p_{ijk}(t) r_{ijk}(t) dt, \quad (1)$$

s.t. $\sum_{i=1}^{I} \int_{0}^{T} p_{ijk}(t) dt \leq c_j, \quad j = 1, 2, ..., J; k = 1, 2, ..., K, \quad (2)$

$\sum_{j=1}^{J} \sum_{k=1}^{K} p_{ijk}(t) \leq \lambda_i(t), \quad i = 1, 2, ..., I, t \in [0, T] \quad (3)$

$p_{ijk}(t) = 0, \quad t = [0, s_{jk} - W) \cup (s_{jk}, T], i = 1, 2, ..., I, \quad (4)$

$\quad j = 1, 2, ..., J, k = 1, 2, ..., K. \quad (5)$
where $k = 1, 2, ..., K$ is the index of a service cycle in the horizon $[0, T]$, and $r_{ijk}(t) = x_i^T A y_j + a(s_{jk} - t) + \epsilon$ is the reward function at arrival time $t$ for an $(i, j, k)$ combination. The decision variable $p_{ijk}(t)$ determines the total expected amount of demand $(i, t)$ that is assigned to the $k$-th occurrence of resource $j$ at time $t$.

We have $V^{OPT} = \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^K \int_0^T p_{ijk}(t) r_{ijk}(t) dt$. The following lemma proves that $V^{OPT}$ provides an upper bound of the optimal dynamic policy.

**Lemma 1.** Let $V^*$ be the expected reward of an optimal dynamic policy that knows the reward function $r(\cdot)$. Then $V^{OPT} \geq V^*$.

**Proof.** Fix any realization of demand represented as a sequence $D = \{(t_d, i_d), D\}$ ($d = 1, 2, ..., D$) where $i_d$ and $t_d$ are the patient type and arrival time of the $d$-th arrival, respectively, and $D$ is a random variable representing the total number of arrivals. Let $p_{i_dj}(t_d) \in [0,1]$ be the assignment probability of resource $(j,k)$ to the $d$-th arrival. Note that the optimal decisions must satisfy

$$
\sum_{d=1}^D p_{i_dj}(t_d) \leq c_j \quad j = 1, 2, ..., J; k = 1, 2, ..., K,
$$

$$
\sum_{j=1}^J \sum_{k=1}^K p_{ijk}(t_d) \leq 1_{i_d=i}, d = 1, 2, ..., D, i = 1, 2, ..., I
$$

$$
p_{i_dj}(t_d) = 0, \quad \forall t_d = [0, s_{jk} - W) \cup (s_{jk}, T].
$$

Since the inequalities hold for any realization $\{(t_d, i_d)\}$ ($d = 1, 2, ..., D$), then it is obvious that the constraints will also be true after taking expectations on both sides. Therefore, the expected optimal offline solution is feasible in problem (1), so that $V^{OPT} \geq V^*$. □

We define the regret as follows:

**Definition 1.** Let $V^{{Alg}}$ be the total expected reward for any algorithm $Alg$. Then the regret for the algorithm is

$$
Reg^{{Alg}} = V^* - V^{{Alg}} \leq V^{OPT} - V^{{Alg}}.
$$

**4. Overview of Algorithm and Main Results**

We will give a brief overview of our algorithm and main results in this section.
4.1. Two-phase algorithm

We first design an algorithm called Two-Phase (TP) that performs exploration and exploitation in two distinct phases. Specifically, the algorithm allocates a period of time of length $T_0$ for exploration. In this phase, patients are randomly assigned to a physician and service slot according to an Exploration Subroutine as shown in (1). Meanwhile, the true rewards are collected to estimate its distribution. We use a Least Square Estimate (LSE) to estimate the reward as a function of the context.

In the exploitation phase $[T_0, T]$, the algorithm optimizes the remaining resource allocation problem based on estimated rewards according to the Marginal Allocation Algorithm in Wang, Truong, and Bank (2015), which is described in (2). One of the challenges is the curse of dimensionality resulting from a large number of resource types, demand types and service slots. Following this algorithm, we first solve a deterministic LP that assigns demands to each resource-type-and-service-slot pair based on aggregate information about demand and capacity. Subsequently, for each resource, we apply the optimal stochastic dynamic policy for admitting demands to that resource.

4.2. Main results

We study the asymptotic regret of the proposed algorithm in the regime that $\lambda_i(t) = \gamma \bar{\lambda}_i(t)$ and $c_j = \gamma \bar{c}_j$, with $\gamma \to \infty$ and $\bar{\lambda}_i(t)$ and $\bar{c}_j$ being fixed units. In the rest of the paper, we use $\bar{x}$ to denote the constant unit of parameter $x$. Let $V^{\text{OPT}}(0, T_0)$ and $V^{\text{OPT}}(T_0, T)$ denote the total expected reward of our algorithm in the exploration and exploitation phase, respectively. Then the regret of our algorithm will be

$$\text{Reg}^{\text{TP}} \leq V^{\text{OPT}} - V^{\text{TP}}(0, T_0) - V^{\text{TP}}(T_0, T)$$

$$= V^{\text{OPT}}(0, T_0) + V^{\text{OPT}}(T_0, T) - V^{\text{TP}}(0, T_0) - V^{\text{TP}}(T_0, T),$$

where $V^{\text{OPT}}(0, T_0)$ and $V^{\text{OPT}}(T_0, T)$ are upper bounds for the total expected rewards of optimal dynamic policy with respect to the capacities in exploration phase and exploitation phase, respectively. Hence, we will bound the regret $V^{\text{OPT}}(0, T_0) - V^{\text{TP}}(0, T_0)$ in exploration phase and $V^{\text{OPT}}(T_0, T) - V^{\text{TP}}(T_0, T)$ in exploitation phase respectively. Our main result is the following theorem:

**Theorem 1.** The regret of TP is

$$\text{Reg}^{\text{TP}} = O((K\gamma)^{\frac{3}{2}}\|\bar{c}\|_1).$$

In addition, the regret of any online algorithm is $\Theta(\|\bar{c}\|_1 K \sqrt{\gamma})$. 
5. Exploration Subroutine

We allocate an interval of length $T_0$ for exploration, that is, to learn the reward function $r(\cdot)$. Specifically, we want to learn the coefficients of the feature weight matrix $A$ and service delay coefficient $a$ given the standard deviation $\sigma$ of the error $\epsilon$.

The challenge in the exploration phase is two-fold. First, we face uncertainty in demand arrivals. Unlike in multi-armed bandit problems, we cannot proactively sample the rewards. Rather, we must wait for customer features, or contexts, to arise. Second, we need to strategically make use of our perishable resources to generate diverse customer-resource pairings, in order to effectively learn the reward as a function of these pairings. Our learning is constrained by the availability of these perishing resources.

Our exploration subroutine first assigns customers to resources with the objective of collecting diverse combinations $(i,j)$, so that we could collect a sample matrix of full rank to conduct least-squares estimation (LSE). Let $\mathcal{M} = \{1, 2, \ldots, M\}$ be the set of independent feature vectors among all $I$ demand types and let $\mathcal{N} = \{1, 2, \ldots, N\}$ be the analogous set of indices for resource types. Note that we have $MN + 1$ independent features including the service delay, it requires at least $MN + 1$ different samples to form a full rank sample matrix.

First, we make sure that at least $MN$ different $(i,j)$ pairs are collected. Secondly we need at least an extra sample with different service delay compared to the rest of the prior samples. For convenience, we define $F_{MN}$ to be the event that $MN$ independent feature vectors are collected when we ignore the service delay as one of the features. We define $F_{MN+1}$ to be the event that at least one extra independent sample vector is collected after $F_{MN}$, with a different service delay than the rest, so that we obtain a full rank sample matrix.

5.1. Precise steps

As mentioned above, our exploration subroutine further separates the exploration phase into two parts. In the first part of length $t_0 = K_0' L$, the subroutine aims to collect all $(i,j), i \in \{1, 2, \ldots, M\}, j \in \{1, 2, \ldots, N\}$, pairings to trigger event $F_{MN}$. In the remaining part of length $t_1 = (K_0 - K_0') L$, the subroutine collects at least one more customer to finally obtain a full rank matrix.

Recall that $t \in [0, T_0]$ is a continuous time index. For each resource, there are in total $K_0 = \frac{T_0}{L} \ (K_0 \in \mathbb{Z}^+)$ recurring service slots within the exploration phase of length $T_0$ and
$K_W = \frac{W}{L}$ recurring service days within each booking window. Given an arrival of customer type $i$ at time $t$, we assign the customer to a resource type $j$ in the $k_w$-th service day in the booking window $(t, t + W]$ with probability $p_{ijkw}$. The assignment probabilities are obtained by solving the following problem:

$$\max_{p_{ijkw}, d_{ij}} \sum_{j=1}^{N} \sum_{i=1}^{M} \log(1 - \sum_{i=1}^{M} (1 - \frac{d_{ij}}{c_j^e})^{K_0 c_j})$$

$$\text{s.t. } d_{ij} \leq \sum_{k_w=1}^{K_W} p_{ijkw} \int_{s_{jk} - W + (k_w - 1)L}^{s_{jk} - W + k_w L} \lambda_i(t) \, dt, i = 1, 2, ..., M,$$

$$j = 1, 2, ..., N, k = 1, 2, ..., K_0$$

$$\sum_{k_w=1}^{K_W} \sum_{j=1}^{N} p_{ijkw} \leq 1, i = 1, 2, ..., M \quad (12)$$

$$\sum_{i=1}^{M} \sum_{k_w=1}^{K_W} p_{ijkw} \int_{s_{jk} - W + (k_w - 1)L}^{s_{jk} - W + k_w L} \lambda_i(t) \, dt \leq c_j, j = 1, 2, ..., N, k = 1, 2, ..., K_0. \quad (13)$$

Here, $k_w = 1, 2, ..., K_W$ is the index of service days in booking horizon $W$. Later in this paper, we will show that the objective of the above optimization problem is the logarithm of the probability of obtaining a full-rank observation matrix. Constraint (11) ensures that the variable $d_{ij}$ captures the least expected number of $(i, j)$ pairings over all occurrences $k = 1, 2, ..., K_0$ of resource $j$. Constraint (12) makes sure that the assignment probabilities for each customer $i$ does not exceed 1. The final constraint ensures that the expected amount of demand assigned to each resource does not exceed the capacity of that resource.

The full exploration subroutine can be described in Algorithm 1.

### 5.2. Scale invariance of exploration subroutine

Next, we show that parameters $p_{ijkw}^*$ and $\frac{d_{ij}^*}{c_j}$ stay unchanged under our scaling of the problem. This implies that the single parameter of problem size $\gamma$ does not effect the exploration policy.
Algorithm 1 Exploration Subroutine.

1: Solve the problem (10) to obtain the assignment probabilities $p_{ijkw}^*$. 

2: while $t < K'_0L$ do 

3: At time $t$, assign any arriving customer $i$ to a resource type $j$ in the $k_w$-th service slot in the booking window $(t, t + W]$ with probability $p_{ijkw}^*$, if the resource has capacity. Otherwise reject the customer. 

4: end while 

5: while $t < T_0$ do 

6: At time $t$, assign any arriving customer $i$ to a resource type $j$ in the $k_w$-th service slot in the booking window $(t, \min \{t + W, T_0\}]$ with probability $p_{ijkw}^*$, if the resource has capacity. Otherwise reject the patient. 

7: end while 

Without loss of generality, we assume that $\lambda_i(t)$ is bounded by some finite constant, so that $\int_{t_1}^{t_2} \lambda_i(t) dt$ exists for any $t_1, t_2 \in [0, T]$. Define 

$$
\lambda_{ijk}(t) = \begin{cases} 
\lambda_i(t)p_{ijKw}, & t \in (s_{jk} - W, s_{j(k+1)} - W] \\
\ldots & \\
\lambda_i(t)p_{ijkw}, & t \in (s_{j(k+k_w-1)} - W, s_{j(k+k_w)} - W] \\
\ldots & \\
\lambda_i(t)p_{ij1}, & t \in (s_{j(k-1)}, s_{jk}] 
\end{cases} 
$$  \hspace{1cm} (14)

\[
\lambda_{ijk}(t) = \begin{cases} 
\lambda_i(t)p_{ijKw}, & t \in (s_{jk} - W, s_{j(k+1)} - W] \\
\ldots & \\
\lambda_i(t)p_{ijkw}, & t \in (s_{j(k+k_w-1)} - W, s_{j(k+k_w)} - W] \\
\ldots & \\
\lambda_i(t)p_{ij1}, & t \in (s_{j(k-1)}, s_{jk}] 
\end{cases} 
\hspace{1cm} (14)
\]

to be the assignment rate of demand $(i, t)$ to resource $(j, k)$. Let 

$$D_{ijk} = \int_{s_{jk} - W}^{s_{jk}} \lambda_{ijk}(t) dt, 
$$  \hspace{1cm} (15)

be the expected total number of arrivals of demand type $i$ to $(j, k)$ over the whole booking horizon. Furthermore, for notational simplification, let 

$$D_{ijkw} = \int_{s_{jk} - W + (k_w - 1)L}^{s_{jk} - W + k_w L} \lambda_i(t) dt 
$$  \hspace{1cm} (16)

be the expected number of arrival of type $i$ demand when $(j, k)$ is the $k_w$-th service slot in booking horizon. Then problem (10) can be rewritten as:

$$
\max_{p_{ijkw},d_{ij}} \sum_{j=1}^{N} \log \left(1 - \sum_{i=1}^{M} (1 - \frac{d_{ij}}{c_{ij}e})^{K'_0} \right) 
$$  \hspace{1cm} (17)
\[ s.t. \quad \frac{d_{ij}}{c_j} \leq \frac{D_{ijk}}{c_j}, \quad i = 1, 2, \ldots, I, \quad j = 1, 2, \ldots, J, \quad k = 1, 2, \ldots, K, \quad (18) \]

\[ \sum_{k_w=1}^{K_W} \sum_{j=1}^{N} p_{ijk_w} \leq 1, \quad i = 1, 2, \ldots, I \quad (19) \]

\[ \sum_{i=1}^{M} \frac{D_{ijk}}{c_j} \leq 1, \quad j = 1, 2, \ldots, J, \quad k = 1, 2, \ldots, K, \quad (20) \]

\[ \frac{D_{ijk}}{c_j} = \sum_{k_w=1}^{K_W} p_{ijk_w} \frac{D_{ijkw}}{c_j}. \quad i = 1, 2, \ldots, I, \quad j = 1, 2, \ldots, J, \quad k = 1, 2, \ldots, K \quad (21) \]

Insert \( D_{ijkw} = \gamma \tilde{D}_{ijkw} \) and \( c_j = \gamma \tilde{c}_j \) into the constraints above, it is easy to see that \( \gamma \) will be omitted from constraints. Therefore, if we scale resource capacity and demand arrival by the same factor \( \gamma \), then according to the reformulation above, the ratio \( \frac{d_{ij}}{c_j} \) and optimal solution \( p^*_{ijkw} \) remain unchanged for any positive \( \gamma \). It implies that given the arrival pattern \( D_{ijkw} \), the exploration policy does not change with problem size \( \gamma \). It could also be proved that the performance for offline policy \( V_{OPT} = \gamma \tilde{V}_{OPT} \).

In the rest of this section, for ease of exposition we assume that the arrival rates are recurrent, that is \( \lambda_i(t) = \lambda_i(t + L) \), so that \( D_{ijk} = D_{ij} \) for all \( k \) and \( D_{ijkw} = D_{ijkw} \).

### 5.3. Estimation error

In this section, we show how to perform estimation by the least squares method, based on the collected samples in the exploration phase. We will analyze the estimation error of our method.

For a pair of vectors \([t, x_i] \in \mathcal{X}^{M+1} \) and \([s_{jk}, y_j] \in \mathcal{Y}^{N+1} \), let \( z \) denote the vector of length \( MN + 1 \) whose \((k-1)N + l\)-th entry is \( x_i[k] \times y_j[l] \), \( k = 1, \ldots, M, \quad l = 1, \ldots, N \), and \( z[MN + 1] = s_{jk} - t \). When resource \((j, k)\) is matched to customer \((t, i)\), \( t \leq s_{jk} \leq t + W \), a reward \( r_{ijk}(t) = \beta^T z_t + \epsilon \) is generated, where \( \beta \) is an unknown vector of length \( MN + 1 \), and the error is \( \epsilon \sim N(0, \sigma^2) \) with known \( \sigma \). Let \( z_n \) be the feature of \( n \)-th collected sample, and we also define \( Z_n \) and \( R_n \) be the sample matrix and reward vector at the time of \( n \)-th arrival.

Let \( Z_0 = \emptyset \), and \( R_0 = \emptyset \) initially. From time 0 to \( T_0 \), we assign resources to customers according to the exploration subroutine. Upon the arrival of \( n \)-th customer, we can compute \( z_n \) and observe \( r_{ijk}(t_n) \). We add \( z_n \) as a row to the end of matrix \( Z_n \), and \( r_{ijk}(t_n) \) to the vector \( R_n \).

Let \( \tau \) be the final number of observations we collect by the time \( T_0 \). Then \( Z_\tau \) is a \( \tau \times (MN + 1) \) matrix and \( R_\tau \) is a vector of length \( \tau \). Recall that in our model \( R_\tau = Z_\tau \beta + \epsilon \), and the Least Square Estimate for \( \beta \) is
\[ \hat{\beta} = (Z^T \tau Z^T \tau)^{-1} Z^T \tau R^T \tau, \]

For \( \hat{\beta} \) to be well-defined, we first need the matrix \( Z^T \tau \) to have rank \( MN + 1 \). In the next section, we will give the probability of event \( F_{MN+1} \) considering that \( \tau \) is a random variable. In this section, we will focus on showing how the estimate converges with the number of samples, given that \( Z^T \tau \) is full ranked.

Under the assumption that \( \epsilon \sim N(0, \sigma^2) \), the estimators for \( \beta \) and \( r \) also follow normal distributions by the following lemma.

**Lemma 2.** Given that \( \epsilon \sim N(0, \sigma^2) \) for a known \( \sigma \), we have:

1. \( \hat{\beta} \) follows the joint normal distribution with mean \( \beta \) and covariance matrix \( \sigma^2(Z^T \tau Z^T \tau)^{-1} \);
2. \( \hat{r} = z^T \hat{\beta} \) follows the normal distribution \( N(z^T \beta, \sigma^2 z^T (Z^T \tau Z^T \tau)^{-1} z) \).

**Proof** Since \( R^T = Z^T \tau + \epsilon \), we have

\[
\hat{\beta} = (Z^T \tau Z^T \tau)^{-1} Z^T \tau R^T \tau = (Z^T \tau Z^T \tau)^{-1} Z^T \tau (Z^T \tau \beta + \epsilon) = \beta + (Z^T \tau Z^T \tau)^{-1} Z^T \tau \epsilon.
\]

Hence \( E[\hat{\beta}] = \beta \), and

\[
Var(\hat{\beta}) = E[(\hat{\beta} - \beta)(\hat{\beta} - \beta)^T] = E[(Z^T \tau Z^T \tau)^{-1} Z^T \tau \epsilon((Z^T \tau Z^T \tau)^{-1} Z^T \tau)^T] = (Z^T \tau Z^T \tau)^{-1} Z^T \tau E[\epsilon \epsilon^T] Z^T \tau (Z^T \tau Z^T \tau)^{-1} = \sigma^2 (Z^T \tau Z^T \tau)^{-1}.
\]

The proof for the second result follows from Theorem 3.8 of Shao (1980). □

The second result in the previous lemma provides us with the following bound for the reward estimate:
Lemma 3. Let $\hat{\beta}$ be the estimator for $\beta$ by the end of $T_0$, and assume that there are totally $\tau$ samples. Let $z$ be a feature vector in feature space, so that the true expected reward is $r = z\beta$, and the estimated value is $\tilde{r} = z\hat{\beta}$. Then we have

$$P\{|r - \tilde{r}|_{\infty} \leq \frac{\eta_1}{\sqrt{\tau}}\} \geq 1 - \alpha. \quad (22)$$

for some constant value $\eta_1$.

Please check appendix A for the proof.

Then we are able to bound the estimate error by lemmas above.

Theorem 2. Let $\mu = K_0 \sum_{j=1}^{N} \sum_{i=1}^{M} D_{ij} (1 - \sum_{l=1}^{M} D_{ij}) - \sqrt{K_0 \sum_{j=1}^{N} \sum_{i=1}^{M} D_{ij}}$, then

$$P\{|r - \tilde{r}|_{\infty} \geq \frac{\eta_1}{\sqrt{\mu}} | F_{MN+1}\} \leq \alpha + \delta_{r}. \quad (23)$$

Proof From Chebyshev’s inequality, we have

$$P(\tau - E[\tau] \geq \sqrt{\frac{\text{Var}[\tau]}{\delta_{r}}}) \leq 1 - \delta_{r}.$$ 

So

$$P(\tau \leq E[\tau] - \sqrt{\frac{\text{Var}[\tau]}{\delta_{r}}}) \leq P(\tau - E[\tau] \geq \sqrt{\frac{\text{Var}[\tau]}{\delta_{r}}}) \leq \delta_{r}.$$ 

Let $\tau_{jk0}$ be the number of all customer types admitted to the $k_0$th resource of type $j$ under the exploration subroutine. Then $E[\tau] = K_0 \sum_{j=1}^{N} E[\tau_{jk0}]$ considering recurrent arrivals. $\tau_{jk0}$ is a truncated Poisson random variable with rate $\lambda_{jk0} = \sum_{i=1}^{M} D_{ij}$, and it cannot exceed the resource capacity $c_j$. Therefore,

$$E[\tau_j] = \sum_{l=1}^{c_j} \frac{\lambda_{j}}{l!} e^{-\lambda_{j}} + \sum_{l=c_j+1}^{\infty} c_j \frac{\lambda_{j}}{l!} e^{-\lambda_{j}}$$

$$= \lambda_{j} \sum_{l=1}^{c_j} \frac{\lambda_{j}}{(l-1)!} e^{-\lambda_{j}} + \sum_{l=c_j+1}^{\infty} c_j \frac{\lambda_{j}}{l!} e^{-\lambda_{j}}$$

$$\geq \lambda_{j} \sum_{l=0}^{c_j-1} \frac{\lambda_{j}}{l!} e^{-\lambda_{j}}$$

$$= \lambda_{j} P(\text{Poisson}(\lambda_{j}) < c_j)$$

$$= \lambda_{j} (1 - P(\text{Poisson}(\lambda_{j}) \geq c_j))$$

$$\geq \lambda_{j} (1 - \frac{\lambda_{j}}{c_j}).$$
where we have used the Markov inequality in the last step. Hence, $E[\tau] \geq K_0 \sum_{j=1}^{N} \lambda_j (1 - \frac{\lambda_j}{\epsilon_j})$.

Since $\tau$ is the sum of those truncated Poisson random variables, its variance is smaller than that of the sum of those un-truncated Poisson random variables. So $\text{Var}[\tau] \leq K_0 \sum_{j=1}^{N} \lambda_j$. Let $\mu = K_0 \sum_{j=1}^{N} \sum_{i=1}^{M} D_{ij} (1 - \frac{\sum_{i=1}^{M} D_{ij}}{\epsilon_j}) - \sqrt{\frac{K_0 \sum_{j=1}^{N} \sum_{i=1}^{M} D_{ij}}{\delta_r}} \leq E[\tau] - \sqrt{\frac{\text{Var}[\tau]}{\delta_r}}$. Then we have

$$P(\tau \leq \mu) \leq P(\tau \leq E[\tau] - \sqrt{\frac{\text{Var}[\tau]}{\delta_r}}) \leq \delta_r.$$ 

Let $F_{MN+1}$ denote the event that the observation matrix $Z$ has full rank $MN + 1$. Then by Lemma 3 we have

$$P\{||r - \hat{r}||_{\infty} \geq \frac{\eta_1}{\sqrt{\mu}} | \tau, F_{MN+1} \} = P\{||r - \hat{r}||_{\infty} \geq \frac{\eta_1}{\sqrt{\mu}} | \tau \geq \mu, F_{MN+1} \} P(\tau \geq \mu)$$

$$+ P\{||r - \hat{r}||_{\infty} \geq \frac{\eta_1}{\sqrt{\mu}} | \tau \leq \mu, F_{MN+1} \} P(\tau \leq \mu)$$

$$\leq P\{||r - \hat{r}||_{\infty} \geq \frac{\eta_1}{\sqrt{\mu}} | \tau \geq \mu, F_{MN+1} \} + P(\tau \leq \mu)$$

$$\leq \alpha + \delta_r.$$

Define constants $\eta_2 = \sum_{j=1}^{N} \sum_{i=1}^{M} D_{ij} (1 - \frac{\sum_{i=1}^{M} D_{ij}}{\epsilon_j})$ and $\eta_3 = \sqrt{\frac{\sum_{j=1}^{N} \sum_{i=1}^{M} D_{ij}}{\delta_r}}$ respectively. Then we have $\mu = \eta_2 K_0 - \eta_3 K_0^{0.5}$. Considering the scaling parameter $\gamma$, we have $\mu = \gamma \eta_2 K_0 - \sqrt{\gamma} \eta_3 K_0$, where $\bar{\eta}_2 = \sum_{j=1}^{N} \sum_{i=1}^{M} \tilde{D}_{ij} (1 - \frac{\sum_{i=1}^{M} \tilde{D}_{ij}}{\epsilon_j})$ and $\bar{\eta}_3 = \sqrt{\frac{\sum_{j=1}^{N} \sum_{i=1}^{M} \tilde{D}_{ij}}{\delta_r}}$.

5.4. The bounds

Recall that to compute the LSE, it is required that the observation matrix $Z$ has rank $MN + 1$. Next we show that this condition is satisfied with high probability.

Recall that $F_{MN}$ is the event that we collect $M \times N$ linearly independent observation vectors in the interval $[0, K_0^{1/2} L]$. Then

$$F_{MN+1} = F_{MN} \cup \{\text{an extra patient arrives during } [K_0^{1/2} L, T_0] \}$$

(24)

is a sufficient event allowing us to compute $(Z^T Z)^{-1}$ to get the LSE estimate for $\hat{\beta}$.

It is easy to see that the probability of collecting an extra patient during $[K_0^{1/2} L, T_0]$ is $1 - \exp(- (K_0 - K_0') D_L)$, where $D_L = \int_0^L \sum_{i=1}^{M} \lambda_i(t) dt$ is the expected number of arrivals during each cycle. Suppose that we want the above probability to be greater than $p_{del} \in (0, 1)$, then we choose $K_0 - K_0'$ as $K_0 - K_0' = \arg \min_x \{x : 1 - \exp(-x D_L) \geq p_{del} \}$. So $K_0 - K_0' = - \frac{\log(1 - p_{del})}{D_L}$.
Next, we focus on calculating $P(F_{MN})$. To that end, define $F_j^i$ to be the event that we collect a single customer-resource pair $(i, j)$; and $F_j$ to be the event that we collect all customer-resource pairs $(i, j)$, $i = 1, 2, \ldots, I$. Since the arrivals to each resource are independent according to our learning subroutine, $P(F_{MN}) = \prod_{j=1}^N P(F_j)$. Thus, the likelihood of $F_j$ can be calculated as

$$P(F_j) = P(\cap_{i=1}^M F_j^i).$$ (25)

Instead of calculating the probability of $\cap_{i=1}^M F_j^i$, we consider the inverse event $\cup_{i=1}^M F_j^{-i}$, where $F_j^{-i}$ is the event that resource $j$ is never assigned to demand $i$. We have

$$P(F_j) = 1 - P(\cup_{i=1}^M F_j^{-i}) \geq 1 - \sum_{i=1}^M P(F_j^{-i}).$$ (26)

For each resource $j$ in service day $k$, we can assume without loss of generality that there are $c_j$ copies of the same resource, each with capacity 1. Then the arrival rate to each copy is $D_{ij}/c_j$. Since customers arriving to each copy are accepted on a first-come-first-served basis, the probability of assigning $i$ to each copy is

$$P^1(F_j^i) \geq P(\text{only customer type } i \text{ arrives}) = \frac{D_{ij}}{c_j} \exp(-\sum_{i=1}^M \frac{D_{ij}}{c_j}).$$ (27)

Since $\sum_{i=1}^M D_{ij} \leq c_j$, then $P^1(F_j^i) \geq \frac{D_{ij}}{c_j} \exp(-\sum_{i=1}^M \frac{D_{ij}}{c_j}) \geq \frac{1}{e}$. Thus the probability of not seeing $(i, j)$, given the capacity $c_j$, is $P^{c_j}(F_j^{-i}) \leq (1 - \frac{D_{ij}}{c_j})^{c_j}$. Therefore given all the service slots, the probability of not seeing $(i, j)$ is

$$P(F_j^{-i}) = \prod_{k=1}^{K'_0} P^{c_j}(F_j^{-i}) \leq (1 - \frac{D_{ij}}{c_j})^{K'_0}.$$ (28)

Then we have

$$P(F_j) \geq 1 - \sum_{i=1}^M (1 - \frac{D_{ij}}{c_j})^{K'_0}.$$ (29)

Finally, we have

$$P(F_{MN}) = \prod_{j=1}^N P(F_j) \geq \prod_{j=1}^N (1 - \sum_{i=1}^M (1 - \frac{D_{ij}}{c_j})^{K'_0}).$$ (30)

We are now ready to prove

**Lemma 4.** Let $p_{ijk}^*$ and $d_{ij}^*$ be an optimal solution to problem (10) or (17). Let the optimal objective value be $P_{MAX} = \sum_{j=1}^N \log(1 - \sum_{i=1}^M (1 - \frac{d_{ij}^*}{c_j} e)^{K'_0})$, then we have

$$P(F_{MN}) \geq \exp(P_{MAX}).$$ (31)
Proof. Since $d_{ij}^* \leq D_{ij}$, it is easy to see that

$$P(F_{MN}) \geq \Pi_{j=1}^{N}(1 - \sum_{i=1}^{M} [(1 - \frac{D_{ij}}{c_{je}})^{K_0' c_{e}}]) \geq \exp(P_{MAX}). \quad (32)$$

\[ \square \]

Using the above lemma, we can write

$$P(F_{MN+1}) \geq \exp(P_{MAX}) p_{del}. \quad (33)$$

Suppose we require the probability of having a full rank matrix $Z$, namely, $P(F_{MN+1})$ to be larger than some threshold $p_f$ (i.e. $p_f = 95\%$), then the exploration phase must be have a certain minimal length.

**Corollary 1.** For a full rank matrix $Z$ to occur with probability at least $p_f$, the exploration phase must have

$$K_0 \geq -\frac{e}{\gamma \min_{i,j} d_{ij}^*} \ln \frac{1 - \left(\frac{p_f}{p_{del}}\right)^{\frac{1}{M}}} + \frac{\log(1 - p_{del})}{\gamma D_L}.$$  

**Proof.** Note that $1 - x \leq e^{-x}$ for $0 \leq x \leq 1$, therefore

$$1 - \frac{d_{ij}^*}{c_{je}} \leq e^{-\frac{d_{ij}^*}{c_{je}}}. \quad (34)$$

Thus $(1 - \frac{d_{ij}^*}{c_{je}})^{K_0' c_{e}} \leq e^{-\frac{d_{ij}^* K_0' c_{e}}{c_{je}}}$, and

$$P(F_{MN+1}) \geq \Pi_{j=1}^{N}(1 - \sum_{i=1}^{M} [(1 - \frac{d_{ij}^*}{c_{je}})^{K_0' c_{e}}]) p_{del} \quad (35)$$

$$\geq \Pi_{j=1}^{N}(1 - \sum_{i=1}^{M} e^{-\frac{d_{ij}^* K_0' c_{e}}{c_{je}}}) p_{del} \quad (36)$$

For exploration phase length $K_0 \frac{e}{\gamma \min_{i,j} d_{ij}^*} \ln \frac{1 - \left(\frac{p_f}{p_{del}}\right)^{\frac{1}{M}} + \log(1 - p_{del})}{\gamma D_L}$, we have

$$P(F_{MN+1}) \geq \Pi_{j=1}^{N}(1 - \sum_{i=1}^{M} e^{-\frac{d_{ij}^* K_0' c_{e}}{c_{je}}}) p_{del} \geq \Pi_{j=1}^{N}(1 - \sum_{i=1}^{M} \frac{1 - \left(\frac{p_f}{p_{del}}\right)^{\frac{1}{M}}}{M}) p_{del}$$

$$= p_f.$$  

\[ \square \]
Combining with the analysis in section 7, we can see that the optimal length of exploration phase can easily satisfy this minimum length that ensure full rank when the problem is of large size.

Next, we set out to find a bound on the proximity of our LSE estimator to the true parameters.

**Theorem 3.**

\[
P\{|r - \tilde{r}| \geq \frac{\eta_1}{\sqrt{\mu}} \} \leq (\alpha + \delta_r - 1)p_f + 1
\]

where \(\eta_1, \eta_2, \text{ and } \eta_3\) are positive constants and \(\mu = \eta_2K_0 - \eta_3K_0^{0.5}\).

**Proof.** From Theorem 2, we get:

\[
P\{|r - \tilde{r}| \geq \frac{\eta_1}{\sqrt{\mu}} \} = P\{|r - \tilde{r}| \geq \frac{\eta_1}{\sqrt{\mu}} | F_{MN+1} \} P(F_{MN+1}) + P\{|r - \tilde{r}| \geq \frac{\eta_1}{\sqrt{\mu}} | \bar{F}_{MN+1} \} P(\bar{F}_{MN+1})
\]

\[
\leq P\{|r - \tilde{r}| \geq \frac{\eta_1}{\sqrt{\mu}} | F_{MN+1} \} P(F_{MN+1}) + P(\bar{F}_{MN+1})
\]

\[
= P\{|r - \tilde{r}| \geq \frac{\eta_1}{\sqrt{\mu}} | F_{MN+1} \} P(F_{MN+1}) + 1 - P(F_{MN+1})
\]

\[
= (P\{|r - \tilde{r}| \geq \frac{\eta_1}{\sqrt{\mu}} | F_{MN+1} \} - 1)P(F_{MN+1}) + 1
\]

\[
\leq (P\{|r - \tilde{r}| \geq \frac{\eta_1}{\sqrt{\mu}} | F_{MN+1} \} - 1)p_f + 1
\]

\[
\leq (\alpha + \delta_r - 1)p_f + 1
\]

\(\Box\)

To aid the analysis of the exploitation phase, we provide a bound on \(|r - \tilde{r}|_\infty\).

**Corollary 2.** Let \(\delta = \frac{m}{\sqrt{d}} = \eta_1(\eta_2K_0 - \eta_3\sqrt{K_0})^{-0.5}\). Then with probability at least \(p_0 = (1 - \alpha - \delta_r)p_f\), we have

\[
|r - \tilde{r}|_\infty \leq \delta
\]

(37)

where \(\eta_1 = z_\alpha \sigma \sqrt{\bar{\tau}_0^b}\), \(\eta_2 = \sum_{j=1}^N \sum_{i=1}^M D_{ij}(1 - \frac{\sum_{i=1}^MD_{ij}}{c_j})\) and \(\eta_3 = \sqrt{\sum_{j=1}^N \sum_{i=1}^MD_{ij} \delta_r}\). More succinctly, with probability at least \(p_0\), we have

\[
|r - \tilde{r}|_\infty \leq \delta = O\left(\frac{1}{\sqrt{\gamma}}\right).
\]
Proof

\[ P(||r - \tilde{r}||_\infty \leq \delta) = 1 - P(||r - \tilde{r}||_\infty \geq \delta) \]
\[ \geq (1 - \alpha - \delta_x)p_f \]
\[ = p_0. \]

(38)
(39)
(40)

Note that the first inequality follows from Theorem 3, and the last equality follows the definition of \( \delta \). Next,

\[ \delta = \eta_1(\gamma \overline{\eta}_2 K_0 - \sqrt{\overline{\gamma} \overline{\eta}_3 \sqrt{K_0}})^{-0.5} \]
\[ \leq \eta_1(\gamma \overline{\eta}_2 K_0 - \gamma \overline{\eta}_3 \sqrt{K_0})^{-0.5} \]
\[ = \eta_1 \gamma^{-0.5}(\overline{\eta}_2 K_0 - \overline{\eta}_3 \sqrt{K_0})^{-0.5}, \]

(41)
(42)
(43)

so that \( \delta = O(\gamma^{-0.5}). \) □

6. Exploitation phase

The exploitation phase spans the interval \((T_0, T]\). In this interval, based on the estimated rewards, resources are dynamically allocated according to an online policy. We adopt the Marginal Allocation algorithm in Wang, Truong, and Bank (2015) as described below. In this section we focus on the regret analysis of the exploitation phase, and we give lower and upper bounds on the regret incurred in this phase.

**Algorithm 2** Marginal Allocation Algorithm

1. Solve the LP (1) to obtain the assignment probabilities \( p_{ijk}^*(t); \)

2. For each resource and service slot combination \((j, k)\), solve for the benefit function \( f_{jk}(t, x) \) for \( x = 1, 2, \ldots, c_j \) and \( t \in (T_0, T]\) by

\[ f'_{jk}(t, x) = - \sum_{i=1}^{I} p_{ijk}^*(t)(r_{ijk}(t) - \nabla f_{jk}(t, x))^+. \]

(44)

3. If at time \( t \), for the current arrival \( i \), \( r_{ijk}(t) < f_{jk}(t, x) \) for all available service slot \((j, k)\) then reject the arrival; otherwise assign the arrival to any \((j, k)\) in the set

\[ \arg\max_{j,k}\{r_{ijk}(t) - f_{jk}(t)\} \text{ (} j,k \text{ is available at time } t \}. \]

(45)
6.1. Asymptotic regret

We compare the online optimization subroutine of \((TP)\) to that of the offline policy \((OFF)\) over the interval \((T_0, T]\). Recall that the regret is defined as

\[
Reg^{TP}(T_0, T) = V^{OFF}(T_0, T) - V^{TP}(T_0, T).
\] (46)

The regret over the exploitation phase can be decomposed as follows

\[
Reg^{TP}(T_0, T) = (V^{OFF}(T_0, T) - \tilde{V}^{OFF}(T_0, T)) + (\tilde{V}^{TP}(T_0, T) - V^{TP}(T_0, T)),
\] (47)

where \(\tilde{V}^{Alg}(t_0, t)\) denotes the total expected reward of any algorithm \(Alg\) in period \((t_0, t_1]\), assuming the estimated reward \(\tilde{r}\), rather than the true vector \(r\).

In the following analysis, we will bound the regret in an asymptotic regime, where the capacity for each resource as well as the mean demand for each customer type approach infinity, i.e. \(c_j = \gamma \bar{c}_j\) and \(\lambda_i(t) = \gamma \bar{\lambda}_i(t)\) for \(\gamma \to \infty\) while \(\bar{c}_j\) and \(\bar{\lambda}_i(t)\) are fixed, \(i = 1, \ldots, I, j = 1, \ldots, J\).

It is easy to see that

\[
V^{OFF}(T_0, T) - \tilde{V}^{OFF}(T_0, T) \leq \frac{T - T_0}{L} \gamma \|\bar{c}\|_1 \|r - \tilde{r}\|_{\infty},
\]

and

\[
\tilde{V}^{TP}(T_0, T) - V^{TP}(T_0, T) \leq \frac{T - T_0}{L} \gamma \|\bar{c}\|_1 \|r - \tilde{r}\|_{\infty},
\]

where \(\|\cdot\|_{\infty}\) and \(\|\cdot\|_1\) denote the \(\infty\)-norm and 1-norm respectively, i.e. the maximum and the sum of components of the argument. Thus, we will focus on analyzing the second of the three differences, namely,

\[
\tilde{V}^{OFF}(T_0, T) - \tilde{V}^{TP}(T_0, T).
\]

We can approximate the continuous problem by a discrete-time model, i.e. dividing the horizon into sufficiently small periods so that at most one arrival will occur in each period. It suffices to bound the regret for one resource at a time. To this end, we will focus on a single resource \(j\) offered in the \(k\)-th cycle, together with its stream of customers. We assume customers arrive at time \(t = 1, \ldots, S(S = T - T_0\text{ as in our case})\). Then when we normalize
the reward vector $r$ to be at most 1, the worst-case regret over all problem instances can be expressed as

$$\max_r \ p \sum_{t=1}^{S} r_t - f_1(c),$$

(48)

s.t.  \quad 0 \geq -f_t(x) + pr_t + pf_{t+1}(x-1) + (1-p)f_{t+1}(x), \ t = 1, 2, ..., S; \ x = 1, 2, ..., c, \ (49)

$$0 \geq -f_t(x) + f_{t+1}(x), \ x = 1, 2, ..., c; \ t = 1, 2, ..., S, \ (50)$$

$$0 \leq r_t \leq 1, \ t = 1, 2, ..., S. \ (51)$$

The first and second set of constraints guarantee that

$$f_t(x) = p \max(r_t + f_{t+1}(k-1), f_{t+1}(k)) + (1-p)f_{t+1}(k).$$

Note that $p$ is a small value resulting from having sufficiently small periods. Following Wang, Truong, and Bank (2015), we homogenize time, such that $p_t = p$ for all $t$. Such transformation has the effect of changing $r(\cdot)$. Therefore, it is without loss of generality.

To make the analysis easier, we also formulate the dual of this problem. Let $\alpha$, $\beta$ and $\omega$ be dual variables corresponding to the first, second and third set of constraints, respectively.

The dual can be written as

$$\min_{\alpha, \beta, \omega} \sum_{t=1}^{S} \omega_t$$

(52)

s.t.  \quad \alpha_{tx} + \beta_{tx} \leq (1-p)\alpha_{t-1x} + p\alpha_{t-1x+1} + \beta_{t-1x}, \ t = 2, ..., S; \ x = 1, 2, ..., c-1, \ (53)

$$\alpha_{tc} + \beta_{tc} \leq (1-p)\alpha_{t-1c} + \beta_{t-1c}, \ t = 2, ..., S, \ (54)$$

$$\alpha_{1x} + \beta_{1x} \leq 0, \ x = 1, 2, ..., c-1, \ (55)$$

$$\alpha_{1c} + \beta_{1c} \leq 1, \ (56)$$

$$\omega_t \geq p(1 - \sum_{x=1}^{c} \alpha_{tx}), \ t = 1, 2, ..., S. \ (57)$$

Based on the dual problem, we can characterize the optimal solutions and values as follows.

**Lemma 5.** The solution $\beta^* = 0$, $\alpha_{tx}^* = \begin{cases} \frac{t-1}{c-x} & \text{for all } x \text{ and } t \text{ satisfying } x+t \geq c+1, \\ 0 & \text{otherwise} \end{cases}$ and $\omega_t^* = p(1 - \sum_{x=1}^{c} \alpha_{tx}^*)$ for all $t$, is optimal for (52).

Please check B for the proof.
6.2. Lower bound for the regret of any online algorithm

Now we will derive a lower bound on the regret of any online algorithm for this problem based on the dual formulation. In particular, we show that no algorithm can achieve a regret that is better than this lower bound.

Recall that the regret of a dynamic resource-allocation policy comes from two sources, one is the estimation error, i.e. $V^{OFF}(T_0, T) - \tilde{V}^{OFF}(T_0, T)$ and the other is from dealing with the uncertainty in demand arrivals, i.e. $\tilde{V}^{OFF}(T_0, T) - \tilde{V}^{TP}(T_0, T)$ and $\tilde{V}^{TP}(T_0, T) - V^{TP}(T_0, T)$. Since the estimation error is greater than or equal to zero, we have that for any algorithm

$$\text{Reg}^{Alg} \geq \tilde{V}^{OFF}(T_0, T) - \tilde{V}^{Alg}(T_0, T).$$

(58)

Recall that in the asymptotic case, we have $c = \gamma \bar{c}$ and $S = \gamma \bar{S}$ since we keep $p$ constant. Using the result in Lemma 5, we can use the optimal value of the dual problem to bound the regret as

$$\tilde{V}^{OFF}(T_0, T) - \tilde{V}^{Alg}(T_0, T) \geq \lim_{\gamma \to \infty} \sum_{t=1}^{\gamma \bar{S}} \omega_t^*$$

$$= \lim_{\gamma \to \infty} \sum_{s=\gamma \bar{c}}^{\gamma \bar{S}-1} p \sum_{x=\gamma \bar{c}}^{s} \binom{s}{x} p^x (1-p)^{s-x}$$

$$= \lim_{\gamma \to \infty} \sum_{s=\gamma \bar{c}}^{\gamma \bar{S}-1} p P(\gamma \bar{c} \leq x \leq s)$$

$$\approx \lim_{\gamma \to \infty} \sum_{s=\gamma \bar{c}}^{\gamma \bar{S}-1} p \Phi^c \left( \frac{s - \gamma \bar{c}}{\sqrt{sp(1-p)}} \right)$$

$$= \lim_{\gamma \to \infty} p \sum_{s=\gamma \bar{c}}^{\gamma \bar{S}-1} \Phi^c(t(s)),$$

where $t(s) = \frac{s - \gamma \bar{c}}{\sqrt{sp(1-p)}}$ and $\Phi^c(t) = P(z > t)$ is the complementary cumulative distribution function of the standard normal distribution. Note that the approximation follows from the fact that when $s$ is a large number, the binomial distribution $Bin(s, p)$ can be approximated by a normal distribution $N(sp, sp(1-p))$. 
Define \( s_1 = \gamma \bar{S} - \eta_0 \sqrt{\bar{c}} \). For some finite \( \eta_0 \) we have \( s_1 \geq \gamma \bar{c} \). So we have

\[
\tilde{V}^{OFF}(T_0, T) - \tilde{V}^{Alg}(T_0, T) \geq \lim_{\gamma \to \infty} p \sum_{s=s_1}^{\gamma \bar{S}} \Phi^c(t(s))
\]

(59)

\[
\geq \lim_{\gamma \to \infty} p(\gamma \bar{S} - s_1)\Phi^c(t(s_1)),
\]

(60)

considering that \( \Phi^c(t(s)) \geq \Phi^c(t(s_1)) \) for all \( s \geq s_1 \). While \( \Phi^c(t(s_1)) = \Phi^c(\frac{\gamma \bar{c} - \gamma \bar{S}p + \eta_0 \sqrt{\bar{c}p(1-p)}}{\sqrt{\gamma \bar{S} - \eta_0 \sqrt{\bar{c}p(1-p)}}}) \to \Phi^c(\frac{\bar{c}}{\sqrt{Sp(1-p)}}) \) when \( \gamma \to \infty \), so that we can regard \( \Phi^c(t(s_1)) \) as a constant. Hence,

\[
\tilde{V}^{OFF}(T_0, T) - \tilde{V}^{Alg}(T_0, T) \geq \lim_{\gamma \to \infty} \sqrt{\gamma \bar{c}} \eta_0 \Phi^c(t(s_1)) = \Theta(\sqrt{\gamma}).
\]

Thus, we have proved that

**Theorem 4.** Any online learning algorithm Alg must have

\[ \text{Reg}^{Alg} = \Theta(\sqrt{\gamma}). \]

### 6.3. Upper bound on the regret of TP in exploitation phase

Now we give an upper bound on \( \tilde{V}^{OFF}(T_0, T) - \tilde{V}^{TP}(T_0, T) \). From the optimal solution of the primal problem, we see that the worst-case regret is achieved when the reward is the same for all customer types, so that the optimal stochastic policy behaves like a first-come-first-served policy. Therefore we will bound the regret for such a policy in the following analysis.

Let \( D \) be the total number of arrival of patients to resource \((j, k)\), and \( E[D] = \gamma \bar{S}p = \gamma \bar{c} \). For a first-come-first-served policy \( FCFS \),

\[
\tilde{V}^{OFF}(T_0, T) - \tilde{V}^{FCFS}(T_0, T) = E[D - \min\{D, \gamma \bar{c}\}].
\]

(61)

Therefore,

\[
\tilde{V}^{OFF}(T_0, T) - \tilde{V}^{TP}(T_0, T) \leq \tilde{V}^{OFF}(T_0, T) - \tilde{V}^{FCFS}(T_0, T)
\]

\[
= E[D - \min\{D, \gamma \bar{c}\}]
\]

\[
= E[D - \min\{D, E[D]\}]
\]

\[
\leq 0.4\sqrt{E[D]}
\]

\[
\leq 0.4\gamma^{0.5} \sqrt{\bar{c}}.
\]

(62)
The last inequality holds because it can be checked numerically that $E[(D - E[D])^+]$ is bounded by $0.4\sqrt{E[D]}$ when $D$ is a large Poisson random variable. Thus, we have proved

**Proposition 1.** For the Two-Phase Algorithm,

$$\hat{V}^{OFF}(T_0, T) - \hat{V}^{TP}(T_0, T) = O(\gamma^{0.5}).$$

### 7. Optimal exploration for TP

In this section, we give an upper bound for the regret of TP over both exploration and exploitation phases. As part of this analysis, we also find the optimal length for the exploration phase.

**Lemma 6.** The total regret of TP over the whole horizon $[0, T]$ is

$$\text{Reg}^{TP} \leq K_0 \gamma \|\bar{c}\|_1 + \gamma^{0.5}\|\bar{c}\|_1 \tilde{\eta}_4(K - K_0)K_0^{-0.5} + 0.4\gamma^{0.5}\|\sqrt{\bar{c}}\|_1(K - K_0),$$

where $\tilde{\eta}_4 = 2\eta_1(\bar{\eta}_2 - \bar{\eta}_3)^{-0.5}$.

Please check appendix C for the proof.

Observe in the total regret that the regret in the exploration phase increases, while that of exploitation phase decreases, with exploration length $K_0$. Let us define the regret as a function of the exploration phase $K_0$ as follows:

$$\text{Reg}(K_0) = K_0 \gamma \|\bar{c}\|_1 + \gamma^{0.5}\|\bar{c}\|_1 \tilde{\eta}_4(K - K_0)K_0^{-0.5} + 0.4\gamma^{0.5}\|\sqrt{\bar{c}}\|_1(K - K_0).$$

The following lemma gives the optimal exploration phase length:

**Lemma 7.** The optimal length for exploration phase that minimize the asymptotic regret is of

$$K_0^* = \left(\frac{\tilde{\eta}_4 K}{2\gamma^{0.5}}\right)^\frac{2}{3}.$$  (63)

Please check Appendix D for the proof. Inserting $K_0^*$ into the regret bound and considering $\gamma \to \infty$, we have

$$\text{Reg}^{TP} = O((\tilde{\eta}_4 K\gamma)^\frac{2}{3}\|\bar{c}\|_1)$$  (64)

This gives the upper bound on the regret in Theorem 1.
8. Computational Experiments

We test our algorithm and theoretical results in this section based on real data from HaoDF, one of the largest online healthcare service platforms in China, where online appointments and online consultations are the most two popular services. Both of these two services share similar service processes in that patients upload information about their conditions in both text and questionnaire format. The service resources are then allocated based on the provided information. Currently, the platform has established effective and efficient disease-prediction algorithms based on Natural Language Processing and Deep Learning, which are used to transform the information uploaded by patients into structured data format, such as disease category and severity level. We focus on two disease types in Dermatology:

leucoderma and acne, and the complete feature vectors for patients in these two disease categories in Table 2.

In addition, we define the physician feature vectors in Table 3. The first two features

are computed from physician ratings by patients. The third feature is measured by the hospital rating, ranging from 1 for the highest ranked hospitals, to 0.8 for lower ranked hospitals. And for the fourth feature, we let the value be 1 for attending physicians and 0.7 for associate attending physicians.
Figure 1  Daily arrival rate for each type of patients during a week.

Based on the features above, we select 5 patient types and 4 physicians to run our numerical experiments. We first randomly generate the reward parameters of matching 5 patient types to 4 physicians from a uniform distribution $U(0.5, 1)$. We chose 0.5 as a lower bound because rewards that are too small ($r < 0.5$) might imply incompatibility between the pair of patient and physician altogether. Given the true reward parameters and features, we next calculate the true feature weights $\beta$ by a linear model. Note that the linearity is merely assumed by our model for the purpose of computation. Thus, where it is true, we expect our algorithm to perform much better. Finally, the arrival rates are calculated separately for each patient type on hourly basis from the real data in Haodf platform. As indicated in the following figures, the arrival rates are non-stationary, varying with both the day of week and hour of day.
We set a week to be a recurrent service cycle, i.e. $L = 1$ week, and assume that the arrival patterns are the same from week to week. We input the arrival data into simulation model. We first solve the offline optimal, and then simulate our $TP$ algorithm by randomly generating arrivals according to the arrival rates. Figure 4 shows the performance of TP algorithm compared to the Offline optimal and the Marginal Allocation Algorithm of Wang et al. (2015). In both the Offline algorithm and MAA, the reward parameters are known to the system. The Offline optimal increases linearly with $\gamma$ since it is obtained by solving the same LP. The performance of MAA is obtained by summing up all the expected value function expression (44) for each $(j, k)$, and that of TP is computed from simulation. It is clear that MAA lies between Offline optimal and TP, and the difference between MAA and TP implies the value of reward parameter information. The regret of TP algorithm is
shown in Figure 4. The average regret is the value of regret divided by scale parameter $\gamma$. In general, the regret is increasing in scale parameter while average regret is opposite.

**Figure 5** The frequency of obtaining a full rank observation matrix $Z$ by our exploration subroutine.

![Figure 5](image)

**Figure 6** The estimation error with scale parameter $\gamma$.

![Figure 6](image)

Other metrics in our algorithm are the probability of obtaining a full rank observation matrix to conduct LSE, and the estimation error. In this numerical experiment, we fix the length of the exploration phase to one cycle, that is $K_0 = 1$, and show how the probability $P(F_{MN+1})$ changes with $\gamma$ in Figure 5. We run our exploration subroutine for 100 replications for each of the $\gamma$ values and count the number of replications in which we get a full rank matrix $Z$. It is obvious that the probability increases with gamma, and this help us to determine an appropriate length of exploration phase to collect data. Figure 6 shows the estimation error versus the scale parameter $\gamma$. The red line indicates the true mean reward, and the gray lines give a 95% confidence interval. The interval is symmetric according to
the definition of confidence interval. Since the offline optimal provides an upper bound for any dynamic learning algorithm, the results demonstrate that our TP algorithm performs quite well in this practical problem, especially in large-scale instances.

References


Hu, Ming, Yun Zhou. 2016. Dynamic type matching.


Wang, Xinshang, Van-Anh Truong, David Bank. 2015. Online advance admission scheduling for services, with customer preferences.
Appendix A: Proof for lemma 3

Proof. According to Lemma 2, \( z\tilde{\beta} \) follows the normal distribution, and the variance is known. So we can construct a confidence interval around \( z\tilde{\beta} \):

\[
z\tilde{\beta} \pm z_{\alpha/2} \frac{\sigma}{\sqrt{T}} \sqrt{Z^T Z - 1} z_z \leq \text{tr}((Z^T Z)_r^{-1})z^T z,
\]

with confidence level \( 1 - \alpha \), where \( z_{\alpha/2} \) is the standard normal distribution coefficient.

Next we show how the width of confidence interval changes with the scaling parameter. As a property of eigenvalues, we have

\[
z^T (Z^T Z)_r^{-1} z \leq \lambda_{\text{max}} z^T z \leq \text{tr}((Z^T Z)_r^{-1})z^T z,
\]

where \( \lambda_{\text{max}} \) is the maximum eigenvalue of \((Z^T Z)_r^{-1}\). By the Sherman-Morrison formula, after adding one more sample \( z' \) into \( Z_r \), the trace of the inverse matrix \((Z^T Z)_r^{-1}\) becomes

\[
\text{tr}((Z_{r+1}^T Z_{r+1})^{-1}) = \text{tr}((Z_r^T Z_r)^{-1}) - \frac{\text{tr}((Z_r^T Z_r)^{-1} z' z'^T (Z_r^T Z_r)^{-1})}{1 + z'^T (Z_r^T Z_r)^{-1} z'},
\]

where the first inequality follows (66), and the second inequality follows from \( \text{tr}(AB) \leq \text{tr}(A)\text{tr}(B) \) so that \( \text{tr}(z' z'^T) = \text{tr}((Z_r^T Z_r)(Z_r^T Z_r)^{-1} z' z'^T) \) \( \leq \text{tr}((Z_r^T Z_r)\text{tr}((Z_r^T Z_r)^{-1} z' z'^T) + \text{tr}((Z_r^T Z_r)) \)). Suppose \( a \leq z'^T z' \leq b \) for all possible \( z' \), then it is obvious that \( \tau a \leq \text{tr}((Z_r^T Z_r)) \leq \tau b \). Let \( x_r = \text{tr}((Z_r^T Z_r)^{-1}) \), then (67) becomes

\[
x_r \leq x_{r-1} - \frac{a}{(\tau - 1)^2 b^2 (1 + bx_{r-1})},
\]

where \( x_{r-1} \) is decreasing, so that \( x_r \leq x_{\alpha} \) for all \( \tau \in [\alpha] \), Then

\[
z_{\alpha} = z_{\alpha/2} \sqrt{x_{\alpha} b}, \text{ so that } P[||r - \hat{r}||_\infty \leq \frac{\eta_1}{\sqrt{\tau}}] \geq 1 - \alpha.
\]

□

Appendix B: Proof for lemma 5

Proof. It’s easy to check that \((\alpha^*, \beta^*, \omega^*)\) is feasible for the dual problem. We will construct a feasible solution for the primal problem and show that it satisfies complementary slackness with \((\alpha^*, \beta^*, \omega^*)\).

Let \( f^*_i(x) = 0 \) for \( t \geq S + 1 \) and \( x \leq 0 \). For \( t = 1, 2, ..., S \), let \( r^*_t = 1 \) and

\[
f^*_i(x) = pr^*_i + pf^*_{i+1}(x - 1) + (1 - p)f^*_{i+1}(x) \geq 0.
\]

We claim that \( f^*_i(x) \geq f^*_{i+1}(x) \) for all \( t = 1, 2, ..., S \) and \( x = 1, 2, ..., c \). Clearly, the claim holds trivially for \( t = S + 1 \), and then it can be proved easily by induction. That is, assuming the claim holds for \( t + 1 \) and then show that

\[
f^*_i(x) - f^*_{i+1}(x) = pr^*_i + pf^*_{i+1}(x - 1) + (1 - p)f^*_{i+1}(x) - (pr^*_{i+1} + pf^*_{i+1}(x - 1) + (1 - p)f^*_{i+1}(x)) \geq 0.
\]
Therefore, the solution is primal feasible.

Then we show \((\alpha^*, \beta^*, \omega^*)\) are complementary slack for the primal feasible solution, that is

\[
\begin{align*}
\alpha^*_i (f_i^*(x) + \beta^*_i f_{i+1}^*(x) + \gamma f_i^*(x) + \delta f_{i+1}^*(x)) &= 0, \\
\beta^*_i (f_i^*(x) + \gamma f_i^*(x) + \delta f_{i+1}^*(x)) &= 0, \\
\omega^*_i (r_i^* - 1) &= 0.
\end{align*}
\]

The last two equalities hold by definition, since \(\beta^* = 0\) and \(r_i^* = 1\), respectively. The first equality holds also by definition, since

\[
f_i^*(x) = pr^*_i + pf^*_i (x - 1) + (1 - p) f^*_i (x).
\]

Therefore, \((\alpha^*, \beta^*, \omega^*)\) is an optimal solution for the dual problem. \(\square\)

**Appendix C: Proof for lemma 6**

*Proof.* We obtain the total regret by summing up the regret in the exploration phase and that in the exploitation phase as follows

\[
\text{Reg}^{TP} = \text{Reg}^{TP}(0, T_0) + (V^{OFF}(T_0, T) - \tilde{V}^{OFF}(T_0, T)) + (\tilde{V}^{OFF}(T_0, T) - V^{TP}(T_0, T))
\]

(74)

The first term is the regret incurred in the exploration phase, and it is smaller than \(K_0 \gamma ||\bar{c}||_1\) since each single resource can only incur regret at most 1.

We now bound the second and last term as follows

\[
\begin{align*}
V^{OFF}(T_0, T) - \tilde{V}^{OFF}(T_0, T) + (V^{TP}(T_0, T) - V^{TP}(T_0, T)) &
\leq 2(K - K_0) \gamma ||\bar{c}||_1 ||r - \bar{r}||_\infty \\
&
\leq 2(K - K_0) \gamma ||\bar{c}||_1 \delta \\
&
\leq 2(K - K_0) \gamma ||\bar{c}||_1 \gamma^{-0.5} \eta_1 (\eta_2 K_0 - \eta_3 \sqrt{K_0})^{-0.5} \\
&
\leq 2(K - K_0) \gamma^{0.5} ||\bar{c}||_1 \eta_1 (\eta_2 K_0 - \eta_3 K_0)^{-0.5} \\
&
\leq 2(K - K_0) \gamma^{0.5} ||\bar{c}||_1 \eta_3 K_0^{-0.5} \\
&
= (K - K_0) \gamma^{0.5} ||\bar{c}||_1 \eta_3 K_0^{-0.5}.
\end{align*}
\]

(76)

By equation (62) and summing up all the regret over all \((j, k)\) combinations, we obtain

\[
\tilde{V}^{OFF}(T_0, T) - V^{TP}(T_0, T) \leq 0.4 \gamma^{0.5} ||\bar{c}||_1 (K - K_0).
\]

\(\square\)

**Appendix D: Proof for lemma 7**

*Proof.* According to the expression of \(\text{Reg}(K_0)\), we have

\[
\frac{\partial \text{Reg}(K_0)}{\partial K_0} = a_1 - a_2 K_0^{-1.5} + a_3 K_0^{-0.5},
\]

(77)
where $a_1 = \gamma \||\tilde{c}||_1 - 0.4\gamma^{0.5}||\sqrt{\tilde{c}}||_1$, $a_2 = 0.5\gamma^{0.5}||\tilde{c}||_1 \bar{\eta}_4 K$, and $a_3 = -0.5\gamma^{0.5}||\tilde{c}||_1 \bar{\eta}_4$. Therefore, to find the root of the above equation, we have to solve a cubic polynomial

$$a_1 x^3 + a_3 x^2 - a_2 = 0, \quad (78)$$

where $x = K_0^{0.5}$. Since the discriminant of the equation $\Delta = -4a_3^2(-a_2) - 27a_1^2(-a_2) < 0$, the equation has only one real root and two conjugate non-real roots.

To solve the cubic polynomial, we transform the equation into another standard form

$$x^3 + px + q = 0, \quad (79)$$

where $p = \frac{-a_2^3}{3a_1^2}$ and $q = \frac{-2a_1a_2^2}{2a_1^3}$. Note that $\frac{\partial(a_1x^3 + a_3x^2 - a_2)}{\partial x} = 3a_1x^2 + 2a_3x > 0$ for $x \geq 0$, because when $\gamma \to \infty$ there are $a_1 > 0 > a_3$ and $|a_1| \gg |a_3|$. So there is only one solution for (78) when $x > 0$, which is

$$x = \sqrt[3]{\frac{-q}{2} + \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}} + \sqrt[3]{\frac{-q}{2} - \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}}. \quad (80)$$

Since $p = O(\frac{1}{\gamma})$ and $q = O(\frac{1}{\gamma^{3/2}})$, when $\gamma \to \infty$ we have $|q| \gg |p|$. Hence, we have an approximate form of the solution, namely $x = \sqrt[3]{-q} = \sqrt[3]{\frac{2a_1^2a_2^2 - 2a_1a_2}{27a_1^3}}$. Furthermore, since $a_1 = O(\gamma)$, $a_2 = O(\sqrt{\gamma})$ and $a_3 = O(\sqrt[3]{\gamma})$, $|a_1| \gg |a_2|$ and $a_1 \gg |a_3|$ when $\gamma \to \infty$. Finally, we have

$$K_0^* = x^2 = \left(\frac{a_2}{a_1}\right)^{\frac{3}{2}} = \left(\frac{0.5\gamma^{0.5}||\tilde{c}||_1 \bar{\eta}_4 K}{\gamma ||\tilde{c}||_1 - 0.4\gamma^{0.5}||\sqrt{\tilde{c}}||_1}\right)^{\frac{3}{2}} = \left(\frac{\bar{\eta}_4 K}{2\gamma^{0.5}}\right)^{\frac{3}{2}}. \quad (81)$$

□