Unanticipated replenishment: online policy for dynamic service composition in manufacturing cloud

Yang HU[a,b], Feng WU[a], Xin LI[b], and Yu YANG[b,*]

[a]School of Management, Xi'an Jiaotong University, Xi'an, Shaanxi, China; [b]School of Data Science, City University of Hong Kong, Hong Kong S.A.R

ARTICLE HISTORY
Compiled August 13, 2024

ABSTRACT
Service composition (SC) a pivotal step in allocating budgeted cloud services to fulfil requests arriving on service platforms. These requests usually arrive in an online pattern, i.e., we do not know which request comes next until the request realises itself. In addition, new services and service providers are eager to join the platform. Thus, a legitimate concern arises: how to allocate services with such service replenishment for the online setting, especially when there is no prior knowledge available for the replenishment or requests? To tackle this challenging problem, we develop a Dual-Price based Online Learning (DPOL) algorithm, and prove that DPOL can achieve a sub-linear regret bound of $O(\sqrt{T})$ against the offline benchmark, where T is the length of the planning horizon. Also, numerical experiments on synthetic datasets validate that the performance of our algorithms outperforms other baseline policies.

KEYWORDS
Cloud manufacturing, online resource allocation, service composition, primal-dual, concentration inequalities

1. Introduction

Cloud manufacturing (CMfg) (Li et al. 2010) revolutionises manufacturing by offering dependable, on-demand, and scalable access to manufacturing services through a pay-as-you-go model (Xu 2012). Within the framework of a typical CMfg workflow, customers submit manufacturing requests to the CMfg platform, encompassing activities such as prototyping, welding, part processing, painting, assembly, and product delivery [1]. Subsequently, the CMfg Platform Operator (PO) engages in a process of matching these requests with corresponding services through Service Composition (SC). This matching process, commonly referred to as Dynamic SC (DSC), entails addressing resource allocation problems within the dynamic cloud environment.

DSC primarily focuses on identifying the optimal policy within a dynamic environment, where customers and service requests are received sequentially. That is to say, the PO has to respond immediately and irrevocably if a request is submitted to the

*Yu YANG is the corresponding author. Please contact email: yuyang@cityu.edu.hk

[1]Readers may refer to INDICS (https://www.casicloud.com/), Amazon Mechanical Turk (https://www.mturk.com/) or COSMOPlat (https://www.cosmoplat.com/) for current manufacturing cloud practice

platform. Slow or delayed response is unwelcome, because customers could become impatient. Therefore, the PO has to make wise trade-off to allocate resources to the current request, or to retain these resources for future profitable requests. Thus, DSC process can also be rendered as an Online Resource Allocation (ORA) problem, where services are budgeted resources with unanticipated replenishment. Allocation decisions often require utilising historical data effectively for successful policy-making, especially for complex systems or unpredictable events. Accurate analysis of historical data is essential to develop efficient allocation strategies, maximising the impact of allocated resources and benefiting the intended recipients.

Meanwhile, a notable new concern arises: for CMfg platforms, especially startups, new service providers constantly join the platform, and new services are updated or published. In such case, maintaining service levels amidst the influx of new service replenishment presents a significant operational challenge. These replenishment, characterised by their exogenous origins and unpredictability, complicate operational planning. The PO is only informed of the volume of services expected at the onset of each discrete time step, necessitating adaptive strategies to ensure platform stability and customer satisfaction.

Regrettably, early policy-making endeavours suggest that evolutionary algorithms (Song et al. 2023; Wang et al. 2022; Hu et al. 2022) are ill-suited for highly responsive environments, while Deep Reinforcement Learning (DRL) approaches (Liu et al. 2022; Wang et al. 2020a) are scrutinised for their subpar learning efficiency, complexity, and performance instability. As to online policies (Zhou et al. 2019; Sumita et al. 2022; Hu, Yang, and Wu 2024; Yang et al. 2024a), the policies referenced in literature assume that budgets are not replenishable, and as such, the total budget constraint is fixed (Balseiro, Lu, and Mirrokni 2023; Lin et al. 2022). It is important to note that these policies make this assumption explicit, as failure to do so renders sub-optimal outcomes. Otherwise, relevant competitive analysis will be invalid. Although some attempts (Yang et al. 2024a; Asgari and Neely 2020; Huang 2020; Qiu et al. 2018) incorporated resource/budget replenishment into their models, they assumed that such replenishment is known independently and identically distributed (k.i.i.d.). But in reality, it is difficult for PO to extract useful rules (or data distributions) from replenishment data, let alone distribution parameters.

Moreover, existing DSC algorithms are primarily designed for single unit consumption (crowdsourcing (Tong et al. 2021), assortment (Zhang et al. 2024), queuing system (Özkan and Ward 2020)); current algorithms cater to specific cases and may not be adaptable to diverse requirements. Others may only be suited for a single request or service type, such as ride-hailing (Zhou et al. 2019; Sumita et al. 2022) or ride-matching. This limits their adaptability and optimal performance in scenarios involving diverse service requirements. Thus, an operation-level comprehensive approach for online platforms is urgently needed.

To address the above challenges, we consider DSC with Unanticipated Replenishment (DSC-UR) problem. We design a Dual-Price based Online Learning (DPOL) algorithm [2] that recognises the importance of treating unanticipated service replenishment differently from the initial inventory. DPOL achieves this by adjusting the dual price of services in each step, which allows the decision maker to strike a balance between reward and dual cost with confidence. As shown in Section IV, proving convergence (presented by Theorem 1) for dual price vector imposes the first technical challenge. Based on The-

[2] *Online* refers to online setting, where customer requests arrive sequentially given limited prior information, see Section II 2.2 for more information about online algorithms and different online research streams; *Learning* indicates DPOL learns dual-price of each type of services in each timestep for decision-making.

orem 1, bounding the *Regret* for the problem becomes the second challenge, since we omit some assumptions that are only for academic discussion. The main contributions of the paper are elaborated as follows

(1) We propose a novel algorithm for the DSC-UR problem. To our knowledge, we are unaware of any previous work addressing DSC-UR problem within DSC context. Even in ORA context, rare previous work is done for average case prohibiting partial- or over-fulfilment. Notably, Yang et al. (2024a) is the most relevant (and most recent) work to our paper. It shares similar settings with our paper. However, different from Yang et al. (2024a), we do not allow partial- or over-fulfilment of customer requests, which means our approach has a nuanced decision space. Further, we do not propose policies for adversarial settings, because worst-case performance is too pessimistic for online platforms. Finally, we do not assume minimum inventory or maximum amount of replenishment, and we use *Regret* for measurement instead of CR (competitive ratio).
(2) Methodologically, we developed DPOL for the case where distribution parameters are not given. DPOL obtains $O(\sqrt{T})$ sub-linear regret bound. Different from Li and Ye (2022), we address the first technical challenge by presenting a first-order Taylor expansion based inequality shown in Lemma 1 facilitating dual-price convergence. The second technical challenge is tackled by presenting two events (shown in Proposition 2) concerning probability constraints.
(3) We conduct numerical experiments to evaluate the performance of DPOL in different contexts. The results validate the theoretical analysis and show the advantage of DPOL.

Outline The remaining paper is organized as follows. In Section II, we review the state-of-the-art on DSC-UR. In Section III, we establish the mathematical basis for DSC-UR. Then, we present the details of DPOL algorithm in Section IV. In Section V, we conduct extensive numerical simulations to evaluate the performance of the proposed algorithm against benchmarks, and Section VI concludes the paper. Proofs and auxiliary results are put in Proof.pdf (supplementary materials) due to limited space.

2. Literature Review

In this section, we briefly review two streams of literature, namely, DSC in the manufacturing cloud, two-sided ORA, and online admission control (OAC) with replenishment. These streams are relevant to our topic's background, methodology and modeling.

2.1. *Dynamic service composition in manufacturing cloud*

Researchers have devoted significant efforts to developing models, algorithms, and frameworks aimed at addressing the challenges of dynamic service composition (DSC) within the context of cloud manufacturing (CMfg), particularly in the presence of uncertainty. Recent progress has led to the introduction of innovative approaches capable of accommodating dynamic services and failures, allowing the utilisation of idle services for compensation. Specifically, both Hu et al. (2022) and Wang et al. (2020a) have investigated a DSC environment characterised by dynamically changing quality of service (QoS) attributes over time. In response to this challenge, Wang et al. (2020a) has developed a prediction-based reinforcement approach to enable better adaptation

to dynamic environments and facilitate the creation of superior service compositions. Furthermore, Liu et al. (2022) has effectively integrated a Deep Deterministic Policy Gradient (DDPG)-based Service Composition (SC) approach within CMfg, thus improving adaptability and scalability. Yang et al. (2024b) tackles two critical challenges in CMfg, say, SC and transportation within hybrid logistic networks. The algorithm incorporates a column generation-based approach for DSC in CMfg for the first time. The authors underscore that the delivery due time, the quality standard, and the discount are significant factors contributing to total cost.

Furthermore, in the pursuit of improved solution quality for evolutionary algorithms within the context of time-varying CMfg SC, Zhou et al. (2022) has leveraged transfer learning to great effect. For a more comprehensive understanding of these advancements, we recommend consulting the latest review on CMfg SC Hayyolalam et al. (2022).

2.2. *Online resource allocation*

ORA is a prominent paradigm for sequential decision-making for a finite horizon. One seminal work of ORA could be traced back to Karp, Vazirani, and Vazirani (1990). They proposed an approximation algorithm attaining the symbolic $1-1/e$ CR for adversarial input, which means that any online variants with more complexity attain no more than $1-1/e$ CR. Technically, the DSC-UR problem shares similarities with vertex arrival two-sided/fully online matching literature to some extent, as both demands and supplies (denoted by vertices) arrive in a dynamic pattern. Wang and Wong (2015) attained the first non-trivial CR of 0.526 and 0.625 hardness, and Tang and Zhang (2022) proved a lower hardness of 0.584 later. Dickerson et al. (2018) presented an early work on an online task assignment with two-sided arrivals. The paper assumed a primitive setting in which only one task and one worker arrive at the system for each time step, and no worker leaves the system unless the worker is assigned to perform a task. Tong et al. (2021) studied a similar but more complicated online task allocation problem in spatial crowdsourcing and developed a TGOA algorithm with attainable constant CR. However, different from our setting, Tong et al. (2021) assumed that the workers' arrival is known and a worker can perform multiple tasks. Moreover, a remarkable variant for two-sided online resource allocation is called online packing (Kesselheim et al. 2014; Vera and Banerjee 2021), where customers require a bundle of multi-dimensional resources. However, a regular online packing setting assumes that resource consumption for each resource is no more than one, and no resources replenishment is considered.

Another stream of literature akin to our problem setting is online multi-item order fulfillment (Jasin and Sinha 2015). A typical setting usually involves multiple items, facilities, regions, and online order types. Jasin and Sinha (2015) proposed the first heuristic of the stream by constructing deterministic linear programming. Amil, Makhdoumi, and Wei (2022) revisited the same problem setting and proposed a hybrid decision-making policy. The policy attains a parameter-dependent CR regardless of the number of items and order types. Unfortunately, Ma (2023) developed enhanced sub-optimal rounding schemes, providing guarantees of $1+\ln(q)$, where q signifies the number of items needed in the order.

2.3. *Online admission control with replenishment*

OAC is another classic online model applied in service systems like online retailing, crowdsourcing, and medical centres. In fact, the DSC-UR problem is more like a multi-

type, multi-item OAC problem: the PO has to *control* whether a service request should be accepted or rejected. In addition, the stochastic process of customers arriving and waiting to be served motivates us to formulate inventory replenishment. Bassamboo, Harrison, and Zeevi (2005) considered a multi-type, multi-item admission control problem with a time-varying customer arrival rate. Customers are either accepted or rejected to serve. Interestingly, even accepted customers may defect (customer loss). Glazebrook, Kirkbride, and Ouenniche (2009) studied a semi-Markov decision process in which customers arrive according to a Poisson process. Also, customers may leave the system if the waiting time is too long. This assumption corresponds to inventory loss, but it is an *anticipated* loss. In a recent study, Legros (2021) examined an OCA problem for a queueing system with state-dependent arrival rates. The findings revealed that the optimal policy takes on a threshold type when the arrival rate exhibits both decreasing and convex characteristics. Feng, Niazadeh, and Saberi (2021) studied an online assortment model. The remarkable thing is that the paper considers exogenous replenishment on inventory, which is inspiring to our paper. Unfortunately, the assortment model differs from ORA.

In summary, we enumerate academic gaps we found in existing works:

(1) Most of the online multi-item order fulfillment papers are unaware of several critical issues in production, such as multi-unit consumption of an item in one order (such case is claimed to be rare (Xu, Allgor, and Graves 2009)), and exogenous resource replenishment during the planning horizon (Acimovic and Graves 2015).
(2) There have been primary discussions on exogenous inventory replenishment under OAC setting already, but such information is usually available, or follows certain distributions by assumption.
(3) Existing literature assumes customer requests usually involve one type of item/resource, but it is apparently impractical. Thus, developing a general model that hybridises multiple request types, multiple items, and multi-unit consumption for online platforms is of great practical value.

3. Problem Formulation

3.1. *Notations*

We include non-negative real (or integer) numbers set for $\mathbb{R}_+$ (or $\mathbb{Z}_+$), whereas n-dimensional $\mathbb{R}_+$ space as $\mathbb{R}_+^n = \{x \in \mathbb{R}^n : x \geq 0\}$. Vectors are denoted by **bold** lowercase letters. For $T \in \mathbb{Z}_+$, $[T]$ is denoted as a shorthand of set $\{1, \ldots, T\}$. Expectation operator and Probability operator are denoted by $\mathbb{E}[\cdot]$ and $\mathbb{P}[\cdot]$. $\mathbb{I}(\cdot)$ is the indicator function, $\mathbb{I}(A) = 1$ if event A is true, and $\mathbb{I}(A) = 0$ otherwise. $(x)^+$ is defined as $\max\{x, 0\}$. For a vector $\boldsymbol{x}$, $\|\boldsymbol{x}\|_1$ denotes its L_1 norm. Relevant parameters, variables and notations with respect to (w.r.t.) our model/algorithm are all presented in Table 1.

3.2. *Problem description*

For a manufacturing cloud, the PO aims to maximize the total reward gained throughout the planning horizon by allocating multiple types of manufacturing services (i.e., resources) to incoming requests. The services are indexed by $i \in [I]$. Each type of service $i \in [I]$ has $C_i \in \mathbb{Z}_+$ units of initial inventory (units) for allocation.

At each timestep $t \in [T]$, at most one type of request arrives. Once arrival, the request must be accepted or rejected immediately and the decision cannot be changed

Table 1. Table of Notations

Notation	Usage
Notation	Usage
T	length of planning horizon
I	Total number of service types
J	Total number of request types
$\mathcal{P}$, $\mathcal{Q}$	Two distributions
$\mathbf{A}$	service consumption matrix
a_{ij}	units of type i service required by type j request
p_i	the dual price for type i service
$\boldsymbol{p}_t$	the dual price vector at timestep t
r_j	the non-negative reward of type j request
j_t	the request type arriving at timestep t
$C_i(t)$	inventory of type i service at the beginning of the timestep t, and $C_i(1) = C_i$
$\boldsymbol{c}$	initial inventory vector divided by T, $(C_1/T, C_2/T, \ldots, C_I/T)^\top$
R_{it}	replenished units of type i service at timestep t
$\boldsymbol{R}_t$	timestep-wise vector for R_{it}
$\boldsymbol{b}$	expectation vector for R_{it}, i.e. $(b_1, ..., b_I)^\top$
b_i	expectation for R_{it}, elements in vector $\boldsymbol{b}$, i.e. $b_i = \mathbb{E}[R_{it}]$
$\bar{R}_i, \underline{R}_i$	upper/lower bound for R_{it}, $i \in [I]$
$\bar{C}, \underline{C}$	upper/lower bound for C_i, $i \in [I]$
$\underline{d}$	a lower bound, $\min_{i\in[I]}(C_i/T + \underline{R}_i)$
x_t	binary decision variable for timestep t
y_t	decision variable for request in timestep t in LP and LP-1
z_t	dual variable appeared in LP-1D
$\bar{a}_i$	maximum units of type i service required within one request
δ	a positive constant
N	a constant depends only on $\bar{R}_i, \underline{R}_i, \bar{a}_i$ and I

thereafter. Different from Jasin and Sinha (2015), we generalise the setting by allowing multiple units of services consumption in one request. If the PO accepts a type j request, it indicates a_{ij} units of type i services are consumed. In practice, one manufacturing requests consumes one or more types of services, as defined by the $\mathbf{A} \in \mathbb{Z}_{I\times J}$ matrix. Rejected requests do not generate any revenue or consume any services.

3.3. *Models*

We rephrase the DSC-UR problem as an online binary program to facilitate our technical discussions. We let $x_t \in \{0, 1\}$ be the binary decision variable for timestep t, where $x_t = 1$ denotes *Accept* the incoming request at t, and $x_t = 0$ denotes *Reject*. The objective of the PO can be expressed as an online stochastic binary program

$$\begin{aligned} & \max_{x_t} \quad \sum_{t=1}^{T} r_{j(t)} x_t \quad \text{(BP)} \\ \text{s.t.} \quad & a_{ij(t)} x_t \le C_i(t) + R_{it} \quad \forall i \in [I], t \in [T] \\ & C_i(t+1) = C_i(t) - a_{ij(t)} x_t + R_{it} \\ & \qquad\qquad\qquad\qquad\qquad\qquad \forall i \in [I], t \in [T-1] \\ & \qquad x_t \in \{0, 1\} \qquad\qquad \forall t \in [T] \end{aligned} \tag{1}$$

In model (1), the first constraint is the inventory constraint, which says that the consumed units should be no more than the current service inventory plus the replenished units for all services and timesteps. The second constraint explains how inventory is updated: on-hand inventory subtracts consumption and then adds replenishment, leading to the updated inventory. The final constraint specifies the value range.

Although we are looking forward to developing a policy that can achieve optimal BP value, calculating the optimal value of (BP) is a daunting task due to its analytical intractability resulting from the curse of dimensionality. Hence, it is more practical to explore an alternative linearized LP instead. The expectation in (LP) serves as the basis for determining the service duration, required service units, and reward.

$$\begin{aligned} &\max_{y_t} \quad \sum_{t=1}^{T} r_{j(t)} y_t \quad \text{(LP)} \\ \text{s.t.} \quad &\sum_{\tau=1}^{t} a_{ij(\tau)} y_\tau \leq C_i + \sum_{\tau=1}^{t} R_{i\tau} \quad \forall i \in [I], t \in [T] \\ &y_t \in [0,1] \qquad\qquad \forall t \in [T] \end{aligned} \tag{2}$$

In LP (2), we first substitute the binary decision variable x_t in (1) with a more relaxed, tractable decision variable $y_t \in [0,1]$ for modeling convenience. Then, we generalise the inventory update into the inventory constraint for a more concise formulation. We also denote vector $\boldsymbol{R}_t = (R_{1t}, R_{2t}, \ldots, R_{It})^\top$ to be the timestep-wise vector for R_{it}.

3.4. *Evaluation metric*

We assume the LP parameters $(r_{j(t)}, \boldsymbol{a}_t)$ and $\boldsymbol{R}_t$ are generated stochastically from two different distributions $\mathcal{P}$ and $\mathcal{Q}$, respectively. However, both $\mathcal{P}$ and $\mathcal{Q}$ are unknown to the PO. We use widely-recognised metric of *Regret* (Li and Ye 2022) to evaluate online algorithms.

The metric *Regret* measures the expected difference between the offline revenue OFF_T and the revenue yield by our proposed policy ALG_T, where offline revenue OFF_T assumes the full knowledge of the realisation (i.e. the request sequence is known in hindsight, which means OFF_T is the optimal revenue to be obtained). *Regret* is defined as follows

$$Regret_T^{\mathcal{P},\mathcal{Q}}(\mathrm{ALG}) = \mathbb{E}_{\mathcal{P},\mathcal{Q}}\left[\mathrm{OFF}_T - \mathrm{ALG}_T\right]$$

4. Dual-price based Online Learning Algorithm

In this section, we present a Dual-Price based Online Learning (DPOL) algorithm for the case when both distributions $\mathcal{P}$ and $\mathcal{Q}$ are unknown (i.e., stochastic setting). Thus, we first present the linear program LP-1

$$\begin{aligned} &\max_{y_t} \quad \sum_{t=1}^{T} r_{j(t)} y_t \quad \text{(LP-1)} \\ \text{s.t.} \quad &\sum_{t=1}^{T} a_{ij(t)} y_t \leq C_i + \sum_{t=1}^{T} R_{it} \quad \forall i \in [I] \\ &y_t \in [0,1] \qquad \forall t \in [T] \end{aligned} \tag{3}$$

It is easy to see LP-1 is derived from LP, the only difference between LP and LP-1 lies in the resource constraints where the constraints are only valid in terms of resource

type i rather than i and timestep t. Given LP-1, we derive the dual formulation of LP-1 below

$$\begin{aligned} \min_{p_i, z_t} \quad & \sum_{i=1}^{I} \left[C_i + \sum_{t=1}^{T} R_{it} \right] p_i + \sum_{t=1}^{T} z_t \quad \text{(LP-1D)} \\ \text{s.t.} \quad & \sum_{i=1}^{I} a_{ij(t)} p_i + z_t \geq r_{j(t)}, \quad \forall t \in [T] \\ & p_i, z_t \geq 0 \qquad \forall i \in [I], t \in [T]. \end{aligned} \tag{4}$$

Here, the decision variables are represented in vector form $\boldsymbol{p} = (p_1, \ldots, p_I)^\top$ and $\boldsymbol{z} = (z_1, \ldots, z_T)^\top$, and let $(\boldsymbol{p}_T^*, \boldsymbol{z}_T^*)$ be an optimal solution for model (4). However, readers may argue if timestep $t \in [T]$ is much more than the number of service type I (i.e., $T \gg I$), how can we yield the solution of LP-1D with reasonable computation cost? In this regard, by substituting the constraints $\sum_{i=1}^{I} a_{ij(t)} p_i + z_t \geq r_{j(t)}$ into the objective function, we omit the variable z_t and obtain a concise form of LP-1D where the solution vector only has dual price $\boldsymbol{p}$.

$$\begin{aligned} \min_{p_i} \quad & \sum_{i=1}^{I} C_i p_i + \sum_{i=1}^{I} \sum_{t=1}^{T} R_{it} p_i + \sum_{t=1}^{T} \left(r_{j(t)} - \sum_{i=1}^{I} a_{ij(t)} p_i \right)^+ \\ \text{s.t.} \quad & p_i \geq 0 \quad \forall i \in [I]. \end{aligned} \tag{5}$$

As we can see, the term $\sum_t \left(r_{j(t)} - \sum_i a_{ij(t)} p_i \right)^+$ is a summation of stochastic functions, and therefore the summation will converge to a certain deterministic function. Model (4) shares an identical optimal solution set with model (5).

Then, in order to yield a final stochastic form, we divide the objective function in (5) by T for an intermediate, Sample Average Approximation (SAA) form (6)

$$\begin{aligned} \min f_T(\boldsymbol{p}) := & \frac{1}{T} \sum_{i=1}^{I} C_i p_i + \frac{1}{T} \sum_{i=1}^{I} \sum_{t=1}^{T} R_{it} p_i + \frac{1}{T} \sum_{t=1}^{T} \left(r_{j(t)} - \sum_{i} a_{ij(t)} p_i \right)^+ \\ \text{s.t.} \quad & p_i \geq 0 \quad \forall i \in [I]. \end{aligned} \tag{6}$$

Finally, based on SAA model (6), we yield the stochastic program shown below

$$\begin{aligned} \min f(\boldsymbol{p}) := & \boldsymbol{c}^\top \boldsymbol{p} + \boldsymbol{b}^\top \boldsymbol{p} + \mathbb{E} \left[\left(r - \boldsymbol{a}^\top \boldsymbol{p} \right)^+ \right] \\ \text{s.t.} \quad & \boldsymbol{p} \geq 0 \end{aligned} \tag{7}$$

here we let $\boldsymbol{c} = (C_1/T, C_2/T, \ldots, C_I/T)^\top$, $\boldsymbol{b} = \mathbb{E}[\boldsymbol{R}_t] = (\mathbb{E}[R_1], ..., \mathbb{E}[R_I])^\top$. This stochastic programming problem re-casts the dual convergence problem. The objective function $f_T(\boldsymbol{p})$ in (6) can be seen as an SAA of the stochastic formulation (7).

4.1. *Details of DPOL algorithm*

In each timestep, DPOL approximates the dual price $\boldsymbol{p}_t^*$ by solving an SAA LP problem with historical observations (line 3-line 4). A request can only be accepted only if the

Algorithm 1 DPOL Algorithm

Input: $\mathbf{A}$, C_i, T, R_{it};
Output: Decision string of $\{Accept, Reject\}$;

1: **for** $t = 1, 2, \ldots, T$ **do**
2: Update inventory $C_i(t)$ with replenishment $R_{it}, \forall i \in [I]$
3: Specify an LP problem

$$\begin{aligned} \arg\min_{\boldsymbol{p}} & \frac{1}{t}\sum_{i=1}^{I}\left(C_i + \sum_{\tau=1}^{t} R_{it}\right)p_i + \frac{1}{t}\sum_{\tau=1}^{t}\left(r_{j(\tau)} - \sum_{i=1}^{I} a_{ij(\tau)}p_i\right)^{+} \\ \text{s.t.} \quad & p_i \geq 0 \qquad \forall i \in [I]. \end{aligned}$$

4: Solve the LP problem above and yield the dual price, $\boldsymbol{p} = (p_1, \ldots, p_I)^{\top}$
5: A type j service request arrives;
6: **if** $r_{j(t)} > \sum_{i=1}^{I} a_{ij}p_i$ and services are enough **then**
7: *Accept* the service request;
8: $C_i(t+1) = C_i(t) - a_{ij}, \forall i \in [I]$;
9: **else**
10: *Reject* the service request;
11: $C_i(t+1) = C_i(t), \forall i \in [I]$;
12: **end if**
13: **end for**

reward is strictly larger than the dual cost (line 6-line 7). Otherwise, DPOL rejects the request.

4.2. *Basic properties for DPOL*

The optimal solutions to both the n-sample approximation model, which is represented by equation (6), and the stochastic model, represented by equation (7), have been denoted as $\boldsymbol{p}_T^*$ and $\boldsymbol{p}^*$, respectively. With $\boldsymbol{p}_T^*$ and $\boldsymbol{p}^*$, we are now able to analyse the convergence of the proposed DPOL algorithm. First, we will present the required assumptions, and then we will formally prove the convergence.

Assumption 1. *We assume the optimal solution $\boldsymbol{p}^*$ to the stochastic optimization problem (7) satisfies $p_i^* = 0$ if and only if $c_i > \mathbb{E}_{(r,\boldsymbol{a})\sim\mathcal{P},\boldsymbol{R}\sim\mathcal{Q}}[a_i \cdot \mathbb{I}(r > \boldsymbol{a}^{\top}\boldsymbol{p}^*) - R_{it}]$.*

Assumption 1 is mild, imposing complementary conditions for the stochastic program (7). Thus, Assumption 1 lays the foundation for presenting Proposition 1

Proposition 1. *Both $f_T(\boldsymbol{p})$ and $f(\boldsymbol{p})$ are convex. Moreover, the optimal solutions $\boldsymbol{p}_T^*$ and $\boldsymbol{p}^*$ satisfy $(\boldsymbol{c}+\boldsymbol{b})^{\top}\boldsymbol{p}^* \leq \bar{r}$ and $(\boldsymbol{c}+\boldsymbol{b})^{\top}\boldsymbol{p}_T^* \leq \bar{r}$, where $\bar{r} = \max_{i\in[I]} r_i$*

Proposition 1 summarizes several fundamental properties pertaining to models (6) and (7). Both the SAA problem and the stochastic program feature convex objective functions. With Assumption 1 and Proposition 1, we formally define the value space for $\boldsymbol{p}$

$$\Xi_p = \left\{\boldsymbol{p} \geq \mathbf{0}, \|\boldsymbol{p}\|_1 \leq \frac{\bar{r}}{\underline{d}}\right\}$$

where $\underline{d} = \min_{i\in[I]}(C_i/T + \underline{R}_i)$. Now, we derive more basic properties for (7) based on Assumption 1. First, define a function h: $\mathbb{R}^I \times \mathbb{R}^I \times \mathbb{R}^{I+1} \to \mathbb{R}$,

$$h(\boldsymbol{p}, \boldsymbol{b}, \boldsymbol{u}) := \boldsymbol{c}^\top \boldsymbol{p} + \boldsymbol{b}^\top \boldsymbol{p} + \left(u_0 - \sum_i u_i p_i\right)^+$$

and function ϕ:

$$\begin{aligned}\phi(\boldsymbol{p}, \boldsymbol{b}, \boldsymbol{u}) &:= \frac{\partial h(\boldsymbol{p}, \boldsymbol{b}, \boldsymbol{u})}{\partial \boldsymbol{p}} \\ &= \boldsymbol{c}^\top + \boldsymbol{b}^\top - (u_1, u_2, ..., u_I)^\top \cdot \mathbb{I}\left(u_0 > \sum_i u_i p_i\right)\end{aligned}$$

where $\boldsymbol{u} = (u_0, u_1, ..., u_I)$ and $\boldsymbol{p} = (p_1, ..., p_I)$. The function ϕ is the partial subgradient of h w.r.t. $\boldsymbol{p}$. As we know

$$f(\boldsymbol{p}) = \mathbb{E}_{\boldsymbol{u}\sim\mathcal{P}, \boldsymbol{R}\sim\mathcal{Q}}\left[h(\boldsymbol{p}, \boldsymbol{b}, \boldsymbol{u})\right]$$

We further define

$$\nabla f(\boldsymbol{p}) = \mathbb{E}\left[\phi(\boldsymbol{p}, \boldsymbol{b}, \boldsymbol{u})\right] \tag{8}$$

The function f may not be differentiable based on the current assumptions, but the definition of $\nabla f(\boldsymbol{p})$ serves as a subgradient for f, as shown in Lemma 1 and Proposition 2.

Lemma 1. *For any $\boldsymbol{p} \geq \boldsymbol{0}$, the following inequality holds*

$$f(\boldsymbol{p}) - f(\boldsymbol{p}^*) \geq \nabla f(\boldsymbol{p}^*)(\boldsymbol{p} - \boldsymbol{p}^*) \tag{9}$$

where the expectation is taken w.r.t. $\boldsymbol{u} \sim \mathcal{P}, \boldsymbol{R} \sim \mathcal{Q}$

Proof. It is easy to see $\nabla f(\boldsymbol{p}^*)(\boldsymbol{p} - \boldsymbol{p}^*)$ can be rendered as a first-order approximation for the function f. □

Lemma 1 represents the difference between $f(\boldsymbol{p})$ and $f(\boldsymbol{p}^*)$ with the subgradient function ϕ. Intuitively, by Taylor expansion, a second-order (and more) term is supposed to be incorporated (Li and Ye 2022). However, in order to avoid strong, impractical assumptions and complex computation, we only present first-order term in Lemma 1.

4.3. *Dual Convergence*

Let us now examine the convergence of $\boldsymbol{p}_T^*$ to $\boldsymbol{p}^*$. The SAA function $f_T(\boldsymbol{p})$ can be represented as

$$f_T(\boldsymbol{p}) = \frac{1}{T}\sum_t h(\boldsymbol{p}, \boldsymbol{R}_t, u_t) \tag{10}$$

where $\boldsymbol{u}_t = (r_{j(t)}, \boldsymbol{a}_t)$ and the function h is as defined earlier. Thus, we yield the following lemma

Lemma 2. *For any $\boldsymbol{p} \in \mathbb{R}^I$, we have the following inequality*

$$f_T(\boldsymbol{p}) - f_T(\boldsymbol{p}^*) \geq \frac{1}{T} \sum_t \phi(\boldsymbol{p}^*, \boldsymbol{R}_t, u_t)^\top (\boldsymbol{p} - \boldsymbol{p}^*) \tag{11}$$

Proof. It is easy to see the subgradient $\nabla f(\boldsymbol{p}^*)$ can be approximated by the sample average subgradient $\frac{1}{T} \sum_t \phi(\boldsymbol{p}^*, \boldsymbol{R}_t, u_t)$ □

In the following section, we will use Lemma 2, a modified version of Lemma 1, to show the convergence of $\boldsymbol{p}_T^*$ to $\boldsymbol{p}^*$. Our aim is to prove that the right-hand side of 2 is concentrated around its expected value, similarly to the right-hand side of 1.

Proposition 2. *We have two events, namely:*

$$\mathcal{E}_1 = \left\{ \left\| \frac{1}{T} \sum_t \phi\left(\boldsymbol{p}^*, \boldsymbol{R}_t, \boldsymbol{u}_t\right) - \nabla f\left(\boldsymbol{p}^*\right) \right\|_2 \leq \epsilon \right\}$$

and

$$\mathcal{E}_2 = \left\{ \left\| \frac{1}{T} \sum_t \boldsymbol{R}_t - \boldsymbol{b} \right\|_2 \leq \epsilon \right\}.$$

The probabilities of the two events are lower-bounded:

$$\mathbb{P}\left(\mathcal{E}_1\right) \geq 1 - 2I \exp\left(-\frac{2T\epsilon^2}{I \max_{i \in [I]} (\bar{R}_i - \underline{R}_i + \bar{a}_i)^2} \right).$$

and

$$\mathbb{P}\left(\mathcal{E}_2\right) \geq 1 - 2I \exp\left(-\frac{2T\epsilon^2}{I \max_{i \in [I]} (\bar{R}_i - \underline{R}_i)^2} \right).$$

Proposition 2 says the sample average partial subgradient $\frac{1}{T} \sum_t \phi(\boldsymbol{p}^*, \boldsymbol{R}_t, u_t)$ stays close to its expectation $\nabla f(\boldsymbol{p}^*)$ (8) evaluated at $\boldsymbol{p}^*$ with high probability. Proposition 2 follows directly from a concentration inequality. The resulting probability bound is independent of the distribution $\mathcal{P}$ and holds for any $\epsilon > 0$. Thus, the proposition indicates that, for event 1, the sample average subgradient concentrates around 0 for binding dimensions and concentrates around a gradient; while for $\mathcal{E}_2$, the sample average vector $\boldsymbol{R}_t$ concentrates around its expectation vector $\boldsymbol{b}$. Sufficiently, only with the two events shown in Proposition 2 are satisfied, i.e. $\mathcal{E}_1 \cap \mathcal{E}_2$, we can assume the following inequality holds

$$\|\boldsymbol{p}^* - \boldsymbol{p}\|_2 \leq \delta\epsilon \tag{12}$$

where δ is a positive constant. The above inequality indicates that the L_2 distance between $\boldsymbol{p}^*$ and $\boldsymbol{p}$ is bounded by $\delta\epsilon$ (with high probability). With the above and then integrating with respect to ϵ, we have the following theorem.

Theorem 1. *Given Assumption 1, there does exist a constant N such that*

$$\mathbb{E}\left[\|\boldsymbol{p}^* - \boldsymbol{p}_T^*\|_2\right] \leq \frac{N}{\sqrt{T}}$$

where N depends only on $\bar{R}_i, \underline{R}_i, \bar{a}_i$ and I

4.4. *Regret upperbound*

Initially, we establish an upper bound for the optimal objective value in the offline scenario. Consider the optimization problem

$$\begin{aligned} \max_{\boldsymbol{p}\geq 0} \quad & \mathbb{E}\left[r\mathbb{I}\left(r > \boldsymbol{a}^\top \boldsymbol{p}\right)\right] \\ \text{s.t.} \quad & \mathbb{E}[\boldsymbol{a}\cdot\mathbb{I}(r > \boldsymbol{a}^\top \boldsymbol{p}^*)] \leq \boldsymbol{c} + \boldsymbol{b} \end{aligned} \tag{13}$$

we can view this formulation as the primal problem (7). To remove constraints, use the Lagrangian of the deterministic formulation by defining specific parameters

$$g(\boldsymbol{p}) = \mathbb{E}\left[r\mathbb{I}\left(r > \boldsymbol{a}^\top \boldsymbol{p}\right) + \left(\boldsymbol{c} + \boldsymbol{b} - \boldsymbol{a}\cdot\mathbb{I}(r > \boldsymbol{a}^\top \boldsymbol{p})\right)^\top \boldsymbol{p}^*\right]$$

where the expectation is taken w.r.t. $(r, \boldsymbol{a}) \sim \mathcal{P}, \boldsymbol{R} \sim \mathcal{Q}$, and $\boldsymbol{p}^*$ is the optimal solution to the stochastic program (7).

Lemma 3. *Under Assumptions 1, we have*

$$\begin{aligned} g(\boldsymbol{p}^*) &\geq g(\boldsymbol{p}) \\ \mathbb{E}\left[\sum_{t=1}^{T} r_{j(t)} y_t^*\right] &\leq T g(\boldsymbol{p}^*) \end{aligned}$$

and

$$g\left(\boldsymbol{p}^*\right) - g(\boldsymbol{p}) \leq \bar{a}\sqrt{I}\mathbb{E}\left[\|\boldsymbol{p}^* - \boldsymbol{p}\|_2\right] \leq \bar{a}N\sqrt{\frac{I}{T}}$$

Lemma 3 provides a deterministic upper bound for the expected optimal objective value in offline settings. Additionally, the instant reward at time t can be approximated using a dual price vector $\boldsymbol{p}_t$ as $g(\boldsymbol{p}_t)$. With the help of Lemma 3, it becomes possible to upper bound the *Regret* of the algorithm at any given time t. Theorem 2 is built upon the foundation laid by Lemma 3, telling us the *Regret* obtained by DPOL algorithm without distribution knowledge of $\mathcal{P}$ and $\mathcal{Q}$ is $O(\sqrt{T})$.

Theorem 2. *With the dual-price-based online learning algorithm specified by Algorithm 1, we have*

$$Regret_T^{\mathcal{P},\mathcal{Q}}(\text{DPOL}) \leq O(\sqrt{T})$$

Remark 1. *Someone may find some similarities on formulation and proof in Li and Ye (2022) with our DPOL. In fact, Li and Ye (2022) considers ORA without replenishment, which means the allocation process may halt if resources are running out. But DSC-UR allocation process never stops, because exogenous resources always replenish the inventory. Thus, we adopt a different formulation and proof, e.g., use first-order approximation in Lemma 1, and propose two special events for Proposition 2.*

5. Numerical Studies

In this section, we re-emphasise the results and the provable regret bounds via numerical studies with synthetic data. All experiments are conducted using Python 3.8, and optimization work is performed with Gurobi 10.0.0 (win64).

Referring to Table C.1 in Vera and Banerjee (2021), there are 15 types of requests and 10 types of services. The arrival probabilities and rewards for requests are consistent with Table C.1 in Vera and Banerjee (2021) unless stated. To be consistent with the focus of this paper, we present a new service consumption matrix $\mathbf{A}$, where various requests can be found. Unless otherwise stated, for all scenarios and cases, replenishment uniformly ranges from 2 to 5 units each timestep, request reward ranges from 0 to 20, and initial inventory for all service is 50.

Moreover, the planning horizon length T is set to be 1000 timesteps. We run 30 simulations for all baseline algorithms at each timestep. Specifically, for one simulation, we first generate a request sequence, and then yield offline revenue OFF_T, then run other baseline algorithms for ALG_T.

To our knowledge, most existing online policies cannot fit in the DSC-UR problem. As to Yang et al. (2024a), roughly adapting its policies to DSC-UR that hinders partial / over-fulfillment is unfair w.r.t. computation performance. In this regard, the policies below are chosen to be the baselines

- Greedy policy. If the platform has an adequate service supply, any request will be accepted. It is widely applied in a variety of testing cases as a baseline policy.
- Bayes policy (Vera and Banerjee 2021) solves a relaxed linear program, then accepts requests by expectation-based acceptance rule. Bayes selector was initially derived for cases with prior information. We also added an information update module for Bayes policy to be used in cases without prior information.
- Proposed DPOL algorithm.

5.1. *Performance comparison under unbalanced service replenishment*

The first experiment considers a scenario in which services are replenished in an unstable, fluctuating way. Specifically, we adjust the service replenishment to make it uniformly range from 2 to 20 units for each timestep. Fig. 1 demonstrates the average revenue of baselines gained against offline benchmark (i.e., normalised) given unbalanced service replenishment. In Fig. 1, the proposed DPOL algorithm outperforms other baselines (approximately 0.8 times OFF_T), achieving the best performance. Greedy policy performs the second, and whose performance curve has a visible gap compared with DPOL. However, Bayes policy reveals an unexpected failure if no distribution information is given.

Remark 2. *Someone may argue why the performance curve for DPOL in Fig. 1 and*

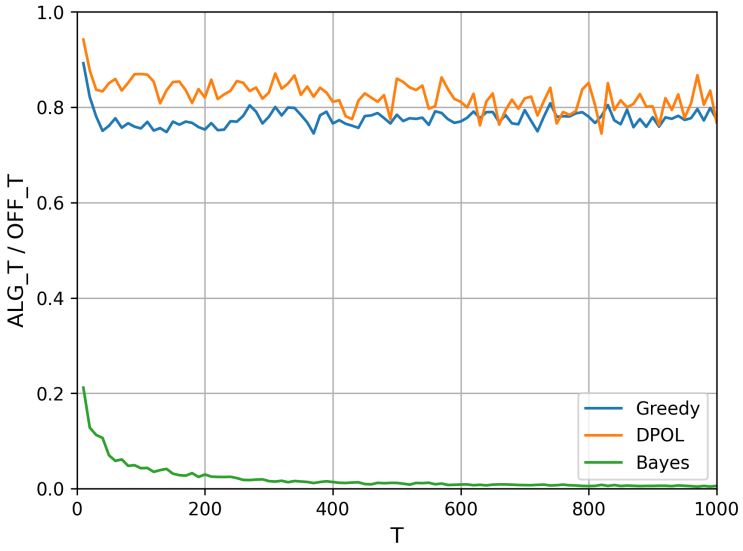

Figure 1. Average revenue gained against offline given unbalanced service replenishment.

2 stabilises near a horizontal level (i.e., the regret scales linearly with T) rather than approximating 1 asymptotically (Theorem 2 states that the regret upper-bound of DPOL increases by $O(\sqrt{T})$ order). A possible reason is the deterministic assignment of reward r_j (Li and Ye 2022).

5.2. *Performance comparison given extreme rewards*

This experiment aims to validate the performance of tasks if there is a significant disparity in rewards. The reward value range is assumed to be vast, ranging from 0 to 500 (rather than 20).

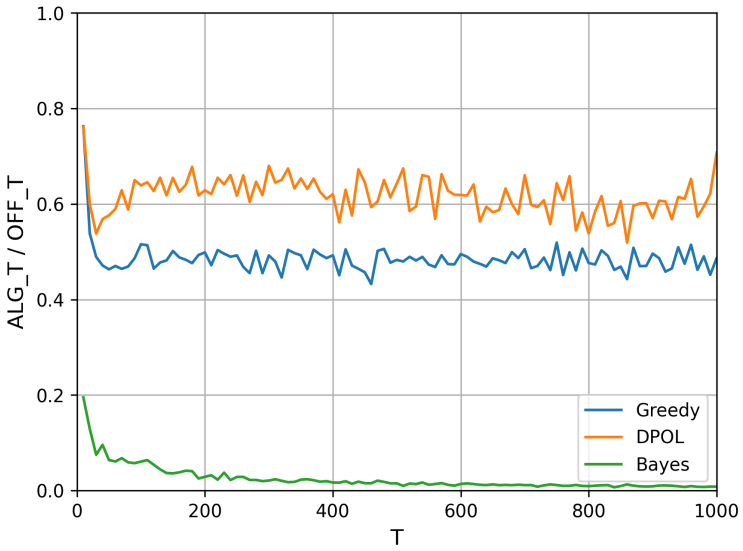

Figure 2. Average revenue gained against offline given extreme rewards.

Table 2. Average reward

Time(/s) \ Policy / T	10	20	50	100	500
5	2235.0 / **2435.7** / 44.7	2001.4 / **2461.9** / 34.4	2642.0 / **3161.6** / 34.1	2064.5 / **2608.2** / 23.8	2746.3 / **3155.1** / 27.8
10	2803.1 / **2964.2** / 45.8	2138.4 / **2588.2** / 23.8	2187.0 / **2796.0** / 21.0	2711.9 / **3248.2** / 27.6	2753.4 / **3254.3** / 38.0
20	**2197.9** / 1633.9 / 33.7	1587.9 / **1993.7** / 24.7	2028.9 / **2539.0** / 29.1	1998.6 / **2489.5** / 30.7	2257.1 / **2843.2** / 27.5
50	**2787.5** / 2262.0 / 36.3	1899.5 / **2081.3** / 25.2	1655.5 / **2301.0** / 21.6	2145.9 / **2761.1** / 16.2	1816.7 / **2377.3** / 20.0
100	1472.6 / **1749.2** / 16.8	1594.9 / **1929.2** / 17.6	2044.6 / **2481.0** / 20.8	1789.3 / **2369.9** / 18.6	1863.6 / **2460.0** / 21.4
200	1789.5 / **1882.0** / 33.8	1795.2 / **2232.1** / 21.5	1670.4 / **2200.5** / 21.3	1678.5 / **2234.4** / 22.7	1729.6 / **2360.2** / 21.0
500	**2265.8** / 1547.2 / 26.7	1645.2 / **1993.8** / 20.0	1912.5 / **2233.8** / 24.6	2037.7 / **2527.8** / 23.5	1747.6 / **2483.2** / 21.0

Notes: The values in each cell stand for the reward obtained by Greedy / DPOL / Bayes. The numbers in bold font denote the best performance for each combination.

Apparently, DPOL algorithm also performs evidently best (approximately 0.6 times OFF_T) over greedy policy (below 0.5 times OFF_T) and Bayes policy (approximately 0) in Fig. 2. DPOL also outperforms the other two benchmarks. Greedy policy performs the second, and whose performance curve has an even larger gap than that in Fig. 1. However, Bayes policy also failed if no distribution information is given. The rigid tuning mechanism in Bayes policy may account for its failure because this mechanism remains too conservative to accept future requests.

5.3. *Performance comparison with increasing types of services and requests*

Table 2 reports the reward of the three policies under different combinations of $I = \{5, 10, 20, 50, 100, 200, 500\}$ and $J = \{10, 20, 50, 100, 200\}$. The estimation is also based on 30 simulations. For each combination of I and J, an instance ($\mathbf{A}$, R_{it} and r_j) is generated from the corresponding model. a_{ij} in $\mathbf{A}$ ranges from 0 to 10, whereas other settings are default. Without loss of generality, we let planning horizon $T = 400$.

Table 2 shows that DPOL performs better than Greedy and Bayes policy in most of the combinations, i.e., $J \geq 20$ and $I \geq 5$. It is expected that DPOL is more effective when I and J are both large. However, we noticed that when $J = 10$, Greedy policy outperforms DPOL with significant advantage given $I = 20, 50, 500$. We can thus infer that fewer request types have adverse effect on DPOL, especially reward r_j is deterministic, relying on request types. Meanwhile, Greedy policy is immune to such effect, because Greedy policy only cares about on-time resource availability. Bayes policy fails again in the experiment with least average reward.

5.4. *Summary*

Based on results shown above, we can make the following conclusions

(1) Bayes policy fails in all cases without prior information. The result suggests that it is unwise to roughly modify any policy requiring information input to no information cases. Methodologically, policies for the two cases are derived from different ideas. However, Bayes policy could possibly outperform other benchmarks with available prior information.
(2) DPOL has best performance given large service replenishment and reward, in which case the dual price mechanism is efficient. However, if service replenishment or inventory is low, the dual price mechanism can hardly take effect.
(3) Greedy policy is the simplest (and fastest) policy with decent performance.

6. Conclusion

We introduce a new variant of the ORA problem, called DSC-UR. This variant is known to have exogenous, unanticipated service replenishment. Moreover, DSC-UR involves multi-item, multi-unit service consumption, and no prior knowledge is provided. To this end, we propose DPOL algorithm for DSC-UR. DPOL is theoretically guaranteed to achieve sub-linear regret bound $O(\sqrt{T})$ against the offline benchmark in average case, where T is the length of the planning horizon. Comprehensive numerical experiments on synthetic datasets validate that DPOL outperforms other baseline policies in most of the cases, e.g., cases with unbalanced service replenishment, extreme rewards or large scale. Our work sheds light on current dynamic matching/service composition/resource allocation problems on the manufacturing cloud.

Acknowledgement

We would like to thank the anonymous reviewers for valuable comments to improve the paper. Yang HU and Feng WU are supported by National Key R&D Program of the Ministry of Science and Technology [Grant number 2018YFB1703001]. Yu YANG is supported partially by the Hong Kong Research Grants Council [ECS 21214720] and City University of Hong Kong [Project 9610465].

Disclosure statement

The authors report there are no competing interests to declare.

Data availability statement

The authors confirm that the data supporting the findings of this study are available within the article and references.

References

Acimovic, Jason, and Stephen C. Graves. 2015. "Making Better Fulfillment Decisions on the Fly in an Online Retail Environment." *Manufacturing & Service Operations Management* 17 (1): 34–51.

Amil, Ayoub, Ali Makhdoumi, and Yehua Wei. 2022. "Multi-item order fulfillment revisited: Lp formulation and prophet inequality." *Available at SSRN 4176274* .

Asgari, Kamiar, and Michael J. Neely. 2020. "Bregman-style Online Convex Optimization with Energy Harvesting Constraints." *Proc. ACM Meas. Anal. Comput. Syst.* 4 (3). https://doi.org/10.1145/3428337.

Balseiro, Santiago R., Haihao Lu, and Vahab Mirrokni. 2023. "The Best of Many Worlds: Dual Mirror Descent for Online Allocation Problems." *Operations Research* 71 (1): 101–119. https://doi.org/10.1287/opre.2021.2242.

Bassamboo, Achal, J Michael Harrison, and Assaf Zeevi. 2005. "Dynamic routing and admission control in high-volume service systems: Asymptotic analysis via multi-scale fluid limits." *Queueing Systems* 51: 249–285.

Dickerson, John P., Karthik Abinav Sankararaman, Aravind Srinivasan, and Pan Xu. 2018. "Assigning Tasks to Workers based on Historical Data: Online Task Assignment with Two-sided Arrivals." In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, AAMAS '18, Richland, SC, 318–326. International Foundation for Autonomous Agents and Multiagent Systems.

Feng, Yiding, Rad Niazadeh, and Amin Saberi. 2021. "Online Assortment of Reusable Resources with Exogenous Replenishment." *SSRN Electronic Journal* 1–37. https://doi.org/10.2139/ssrn.3795056.

Glazebrook, K. D., C. Kirkbride, and J. Ouenniche. 2009. "Index Policies for the Admission Control and Routing of Impatient Customers to Heterogeneous Service Stations." *Operations Research* 57 (4): 975–989. https://doi.org/10.1287/opre.1080.0632.

Hayyolalam, Vahideh, Behrouz Pourghebleh, Mohammad Reza Chehrehzad, and Ali Asghar Pourhaji Kazem. 2022. "Single-objective service composition methods in cloud manufacturing systems: Recent techniques, classification, and future trends." *Concurrency and Computation: Practice and Experience* 34 (5): e6698. https://doi.org/10.1002/cpe.6698.

Hu, Yang, Feng Wu, Yu Yang, and Yongkui Liu. 2022. "Tackling temporal-dynamic service composition in cloud manufacturing systems: A tensor factorization-based two-stage approach." *J Manuf Syst* 63: 593–608. https://doi.org/10.1016/j.jmsy.2022.05.008.

Hu, Yang, Yu Yang, and Feng Wu. 2024. "Dynamic cloud manufacturing service composition with re-entrant services: an online policy perspective." *Int. J. Prod. Res.* 62 (9): 3263–3287. https://doi.org/10.1080/00207543.2023.2230317.

Huang, Longbo. 2020. "Fast-Convergent Learning-Aided Control in Energy Harvesting Networks." *IEEE Transactions on Mobile Computing* 19 (12): 2793–2803. https://doi.org/10.1109/TMC.2019.2936344.

Jasin, Stefanus, and Amitabh Sinha. 2015. "An LP-Based Correlated Rounding Scheme for Multi-Item Ecommerce Order Fulfillment." *Operations Research* 63 (6): 1336–1351. https://doi.org/10.1287/opre.2015.1441.

Karp, R. M., U. V. Vazirani, and V. V. Vazirani. 1990. "An Optimal Algorithm for On-Line Bipartite Matching." In *Proceedings of the Twenty-Second Annual ACM Symposium on Theory of Computing*, STOC '90, New York, NY, USA, 352–358. Association for Computing Machinery.

Kesselheim, Thomas, Andreas Tönnis, Klaus Radke, and Berthold Vöcking. 2014. "Primal beats dual on online packing LPs in the random-order model." In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, New York, NY, USA, 303–312. Association for Computing Machinery.

Legros, Benjamin. 2021. "Dimensioning a queue with state-dependent arrival rates." *Computers & Operations Research* 128: 105179.

Li, B., L. Zhang, S. Wang, F. Tao, and X. Chai. 2010. "Cloud manufacturing:a new service-oriented networked manufacturing model." *Computer Integrated Manufacturing Systems* 16 (1): 1–7+16.

Li, Xiaocheng, and Yinyu Ye. 2022. "Online Linear Programming: Dual Convergence, New Algorithms, and Regret Bounds." *Operations Research* 70 (5): 2948–2966. https://doi.org/10.1287/opre.2021.2164.

Lin, Qiulin, Yanfang Mo, Junyan Su, and Minghua Chen. 2022. "Competitive Online Optimization with Multiple Inventories: A Divide-and-Conquer Approach." *Proc. ACM Meas. Anal. Comput. Syst.* 6 (2). https://doi.org/10.1145/3530902.

Liu, Yongkui, Huagang Liang, Yingying Xiao, Haifeng Zhang, Jingxin Zhang, Lin Zhang, and Lihui Wang. 2022. "Logistics-involved service composition in a dynamic cloud manufacturing environment: A DDPG-based approach." *Robotics & Computer-Integrated Manufacturing* 76: 102323. https://doi.org/10.1016/j.rcim.2022.102323.

Ma, Will. 2023. "Order-Optimal Correlated Rounding for Fulfilling Multi-Item E-Commerce Orders." *Manufacturing & Service Operations Management* 25 (4): 1324–1337.

Özkan, Erhun, and Amy R. Ward. 2020. "Dynamic Matching for Real-Time Ride Sharing." *Stochastic Systems* 10 (1): 29–70. https://doi.org/10.1287/stsy.2019.0037.

Qiu, Chengrun, Yang Hu, Yan Chen, and Bing Zeng. 2018. "Lyapunov Optimization for Energy Harvesting Wireless Sensor Communications." *IEEE Internet of Things Journal* 5 (3): 1947–1956. https://doi.org/10.1109/JIOT.2018.2817590.

Song, Chunhe, Haiyang Zheng, Guangjie Han, Peng Zeng, and Li Liu. 2023. "Cloud Edge Collaborative Service Composition Optimization for Intelligent Manufacturing." *IEEE Transactions on Industrial Informatics* 19 (5): 6849–6858. https://doi.org/10.1109/TII.2022.3208090.

Sumita, Hanna, Shinji Ito, Kei Takemura, Daisuke Hatano, Takuro Fukunaga, Naonori Kakimura, and Ken-ichi Kawarabayashi. 2022. "Online Task Assignment Problems with Reusable Resources." *Proceedings of the AAAI Conference on Artificial Intelligence* 36 (5): 5199–5207. https://doi.org/10.1609/aaai.v36i5.20455.

Tang, Zhihao Gavin, and Yuhao Zhang. 2022. "Improved Bounds for Fractional Online Matching Problems." .

Tong, Yongxin, Yuxiang Zeng, Bolin Ding, Libin Wang, and Lei Chen. 2021. "Two-Sided Online Micro-Task Assignment in Spatial Crowdsourcing." *IEEE Transactions on Knowledge and Data Engineering* 33 (5): 2295–2309. https://doi.org/10.1109/TKDE.2019.2948863.

Vera, Alberto, and Siddhartha Banerjee. 2021. "The Bayesian Prophet: A Low-Regret Framework for Online Decision Making." *Management Science* 67 (3): 1368–1391. https://doi.org/10.1287/mnsc.2020.3624.

Wang, H., J. Li, Q. Yu, T. Hong, J. Yan, and W. Zhao. 2020a. "Integrating recurrent neural networks and reinforcement learning for dynamic service composition." *Future Generation Computer Systems* 107: 551–563. https://doi.org/10.1016/j.future.2020.02.030.

Wang, Min, Shanchen Pang, Shihang Yu, Sibo Qiao, Xue Zhai, and Hao Yue. 2022. "An Optimal Production Scheme for Reconfigurable Cloud Manufacturing Service System." *IEEE Transactions on Industrial Informatics* 18 (12): 9037–9046. https://doi.org/10.1109/TII.2022.3169979.

Wang, Yajun, and Sam Chiu-wai Wong. 2015. "Two-sided Online Bipartite Matching and Vertex Cover: Beating the Greedy Algorithm." In *Automata, Languages, and Programming*, edited by Magnús M. Halldórsson, Kazuo Iwama, Naoki Kobayashi, and Bettina Speckmann, Berlin, Heidelberg, 1070–1081. Springer Berlin Heidelberg.

Xu, Ping Josephine, Russell Allgor, and Stephen C. Graves. 2009. "Benefits of Reevaluating Real-Time Order Fulfillment Decisions." *Manufacturing & Service Operations Management* 11 (2): 340–355.

Xu, Xun. 2012. "From cloud computing to cloud manufacturing." *Robotics and Computer-Integrated Manufacturing* 28 (1): 75–86. https://doi.org/10.1016/j.rcim.2011.07.002.

Yang, Jianyi, Pengfei Li, Mohammad Jaminur Islam, and Shaolei Ren. 2024a. "Online Allocation with Replenishable Budgets: Worst Case and Beyond." *Proc. ACM Meas. Anal. Comput. Syst.* 8 (1). https://doi.org/10.1145/3639030.

Yang, Zhiyuan, Zheyi Tan, Lu Zhen, Nianzu Zhang, Lilan Liu, and Tianyi Fan. 2024b. "Column generation for service assignment in cloud-based manufacturing." *Computers & Operations Research* 161: 106436.

Zhang, Hongbin, Qixin Zhang, Feng Wu, and Yu Yang. 2024. "Dynamic Assortment Selection Under Inventory and Limited Switches Constraints." *IEEE Transactions on Knowledge and Data Engineering* 36 (3): 1056–1068. https://doi.org/10.1109/TKDE.2023.3301649.

Zhou, Jiajun, Liang Gao, Chao Lu, and Xifan Yao. 2022. "Transfer learning assisted batch optimization of jobs arriving dynamically in manufacturing cloud." *Journal of Manufacturing Systems* 65: 44–58. https://doi.org/10.1016/j.jmsy.2022.08.003.

Zhou, Yu-Hang, Chen Liang, Nan Li, Cheng Yang, Shenghuo Zhu, and Rong Jin. 2019. "Robust online matching with user arrival distribution drift." In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33, 459–466.

Appendix A. Auxiliary models

Here, we provide the necessary models for analysis purposes.

A.1. *Hoeffding's inequality*

Let $X_1, \ldots, X_T$ be independent random variables such that X_i takes its values in $[a_i, b_i]$ almost surely for all $i \leq T$. Then for every $\epsilon > 0$,

$$\mathbb{P}\left(\left|\frac{1}{T}\sum_{i=1}^{T} X_i - \mathbb{E}[X_i]\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{2T^2\epsilon^2}{\sum_{i=1}^{T}(a_i - b_i)^2}\right)$$

Hoeffding's inequality regulates the upper bound of the empirical error.

Appendix B. Proofs

B.1. *Proof of Proposition 1*

Proof. Recall the objective function $\min_{p_i} \sum_i C_i p_i + \sum_{i,t} R_{it} p_i + \sum_t \left(r_t - \sum_i a_{ij(t)} p_i\right)^+$, the first and the second summation term are linear, and each component in the third summation is convex. The summation operation preserves convexity. Thus, both f_T and f are convex functions. Moreover, if we assume $(\boldsymbol{c} + \boldsymbol{b})^\top \boldsymbol{p} > \bar{r}$, then

$$f(\boldsymbol{p}) = \boldsymbol{c}^\top \boldsymbol{p} + \boldsymbol{b}^\top \boldsymbol{p} + \mathbb{E}\left[\left(r_t - \boldsymbol{a}^\top \boldsymbol{p}\right)^+\right] \geq (\boldsymbol{c} + \boldsymbol{b})^\top \boldsymbol{p} > \bar{r} \geq \mathbb{E}[r_t] = f(\boldsymbol{0})$$

Hence $\boldsymbol{p}$ cannot be the optimal solution. Thus we have our required result. □

B.2. *Proof of Proposition 2*

Proof. For $\mathcal{E}_1$, recall that the partial derivative function

$$\phi\left(\boldsymbol{p}^*, \boldsymbol{R}_t, \boldsymbol{u}_t\right) = \boldsymbol{c}^\top + \boldsymbol{b}^\top - (u_1, u_2, ..., u_I)^\top \cdot \mathbb{I}\left(u_0 > \sum_i u_i p_i\right)$$

We use $\phi\left(\boldsymbol{p}^*, \boldsymbol{R}_t, \boldsymbol{u}_t\right)_i$ to denote the i-th coordinate of the gradient vector $\phi\left(\boldsymbol{p}^*, \boldsymbol{R}_t, \boldsymbol{u}_t\right)$. By the definition of ϕ, we know that

$$\mathbb{E}\left[\phi\left(\boldsymbol{p}^*, \boldsymbol{R}_t, \boldsymbol{u}_t\right)_i\right] = \left(\nabla f\left(\boldsymbol{p}^*\right)\right)_i.$$

For all types of services, service consumption $\boldsymbol{a}_{j(t)}$ for each timestep is bounded, and we have

$$\phi\left(\boldsymbol{p}^*, \boldsymbol{R}_t, \boldsymbol{u}_t\right)_i \in \left[C_i/T + \underline{R}_i - \bar{a}_i, C_i/T + \bar{R}_i\right].$$

Then, by applying the Hoeffding's inequality, we obtain

$$\mathbb{P}\left(\left|\frac{1}{T}\sum_{t=1}^{T}\phi\left(\boldsymbol{p}^*,\boldsymbol{R}_t,\boldsymbol{u}_t\right)_i-\left(\nabla f\left(\boldsymbol{p}^*\right)\right)_i\right|\geq\epsilon\right)\leq 2\exp\left(-\frac{2T^2\epsilon^2}{\sum_{t=1}^{T}(\bar{R}_i-\underline{R}_i+\bar{a}_i)^2}\right)$$

In fact, event $\mathcal{E}_1$ can be covered by the union of element-wise sub-events as follows

$$\left\{\left\|\frac{1}{T}\sum_{t}\phi\left(\boldsymbol{p}^*,\boldsymbol{R}_t,\boldsymbol{u}_t\right)-\nabla f\left(\boldsymbol{p}^*\right)\right\|_2\geq\epsilon\right\}\subset\bigcup_{i=1}^{I}\left\{\left|\frac{1}{T}\sum_{t}\phi\left(\boldsymbol{p}^*,\boldsymbol{R}_t,\boldsymbol{u}_t\right)_i-\left(\nabla f\left(\boldsymbol{p}^*\right)\right)_i\right|\geq\frac{\epsilon}{\sqrt{I}}\right\}.$$

Applying the union bound, the probability of $\mathcal{E}_1$ is bounded as

$$\begin{aligned}\mathbb{P}\left(\left\|\frac{1}{T}\sum_{t}\phi\left(\boldsymbol{p}^*,\boldsymbol{R}_t,\boldsymbol{u}_t\right)-\nabla f\left(\boldsymbol{p}^*\right)\right\|_2\geq\epsilon\right)&\leq I\mathbb{P}\left(\left|\frac{1}{T}\sum_{t}\phi\left(\boldsymbol{p}^*,\boldsymbol{R}_t,\boldsymbol{u}_t\right)_i-\left(\nabla f\left(\boldsymbol{p}^*\right)\right)_i\right|\geq\frac{\epsilon}{\sqrt{I}}\right)\\&\leq 2I\exp\left(-\frac{2T\epsilon^2}{I\max\limits_{i\in[I]}(\bar{R}_i-\underline{R}_i+\bar{a}_i)^2}\right).\end{aligned}$$

For $\mathcal{E}_2$, we also apply the Hoeffding's inequality, and obtain

$$\begin{aligned}\mathbb{P}\left(\left|\frac{1}{T}\sum_{t=1}^{T}R_{it}-b_i\right|\geq\frac{\epsilon}{\sqrt{I}}\right)&\leq 2\exp\left(-\frac{2T^2\epsilon^2}{I\sum_{t=1}^{T}(\bar{R}_i-\underline{R}_i)^2}\right)\\&=2\exp\left(-\frac{2T\epsilon^2}{I(\bar{R}_i-\underline{R}_i)^2}\right)\end{aligned}$$

then,

$$\left\{\left\|\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{B}_t-\boldsymbol{b}\right\|_2\geq\epsilon\right\}\subset\bigcup_{i=1}^{I}\left\{\left|\frac{1}{T}\sum_{t=1}^{T}R_{it}-b_i\right|\geq\frac{\epsilon}{\sqrt{I}}\right\}.$$

we have

$$\mathbb{P}\left(\left\|\frac{1}{T}\sum_{t=1}^{T}\boldsymbol{B}_t-\boldsymbol{b}\right\|_2\geq\epsilon\right)\leq 2I\exp\left(-\frac{2T\epsilon^2}{I\max\limits_{i\in[I]}(\bar{R}_i-\underline{R}_i)^2}\right)$$

Thus we obtain the Proposition. □

B.3. *Proof of Lemma 3*

Proof. First, we show $g(\boldsymbol{p}^*)$ provides an upper bound for $\mathbb{E}\left[\sum_{t=1}^T r_t y_t^*\right]$:

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^T r_t y_t^*\right] &= \mathbb{E}\left[\boldsymbol{C}^\top \boldsymbol{p}_T^* + \sum_{t=1}^T \boldsymbol{R}_t^\top \boldsymbol{p}_T^* + \sum_{t=1}^T \left(r_t - \boldsymbol{a}_{j(t)}^\top \boldsymbol{p}_T^*\right)^+\right] \\ &\leq \boldsymbol{C}^\top \boldsymbol{p}^* + T\boldsymbol{b}^\top \boldsymbol{p}^* + \mathbb{E}\left[\sum_{t=1}^T \left(r_t - \boldsymbol{a}_{j(t)}^\top \boldsymbol{p}^*\right)^+\right] \\ &= T g(\boldsymbol{p}^*)\end{aligned}$$

Then, by taking the difference between $g(\boldsymbol{p}^*)$ and $g(\boldsymbol{p})$:

$$\begin{aligned}& g(\boldsymbol{p}^*) - g(\boldsymbol{p}) \\ =&\mathbb{E}\left[r\mathbb{I}\left(r > \boldsymbol{a}^\top \boldsymbol{p}^*\right) + \left(\boldsymbol{c} + \boldsymbol{b} - \boldsymbol{a}\cdot \mathbb{I}(r > \boldsymbol{a}^\top \boldsymbol{p}^*)\right)^\top \boldsymbol{p}^*\right] - \mathbb{E}\left[r\mathbb{I}\left(r > \boldsymbol{a}^\top \boldsymbol{p}\right) + \left(\boldsymbol{c} + \boldsymbol{b} - \boldsymbol{a}\cdot \mathbb{I}(r > \boldsymbol{a}^\top \boldsymbol{p})\right)^\top \boldsymbol{p}^*\right] \\ =&\mathbb{E}\left[(r - \boldsymbol{a}^\top \boldsymbol{p}^*)\left(\mathbb{I}\left(r > \boldsymbol{a}^\top \boldsymbol{p}^*\right) - \mathbb{I}\left(r > \boldsymbol{a}^\top \boldsymbol{p}\right)\right)\right] \\ =&\mathbb{E}\left[(r - \boldsymbol{a}^\top \boldsymbol{p}^*)\mathbb{I}\left(\boldsymbol{a}^\top \boldsymbol{p} \geq r > \boldsymbol{a}^\top \boldsymbol{p}^*\right) - (r - \boldsymbol{a}^\top \boldsymbol{p}^*)\mathbb{I}\left(\boldsymbol{a}^\top \boldsymbol{p}^* \geq r > \boldsymbol{a}^\top \boldsymbol{p}\right)\right] \geq 0\end{aligned}$$

This proves the maximum of $g(\boldsymbol{p})$ is achieved at $\boldsymbol{p}^*$. Furthermore, with a more careful analysis, we have

$$\begin{aligned}& g\left(\boldsymbol{p}^*\right) - g(\boldsymbol{p}) \\ =&\mathbb{E}\left[\left(\boldsymbol{a}^\top \boldsymbol{p}^* - r\right)\mathbb{I}\left(\boldsymbol{a}^\top \boldsymbol{p}^* \geq r > \boldsymbol{a}^\top \boldsymbol{p}\right)\right] + \mathbb{E}\left[\left(r - \boldsymbol{a}^\top \boldsymbol{p}^*\right)\mathbb{I}\left(\boldsymbol{a}^\top \boldsymbol{p}^* < r \leq \boldsymbol{a}^\top \boldsymbol{p}\right)\right] \\ \leq&\mathbb{E}\left[\left(\boldsymbol{a}^\top \boldsymbol{p}^* - \boldsymbol{a}^\top \boldsymbol{p}\right)\mathbb{I}\left(\boldsymbol{a}^\top \boldsymbol{p}^* \geq r > \boldsymbol{a}^\top \boldsymbol{p}\right)\right] + \mathbb{E}\left[\left(\boldsymbol{a}^\top \boldsymbol{p} - \boldsymbol{a}^\top \boldsymbol{p}^*\right)\mathbb{I}\left(\boldsymbol{a}^\top \boldsymbol{p}^* < r \leq \boldsymbol{a}^\top \boldsymbol{p}\right)\right] \\ =&\mathbb{E}\left[\boldsymbol{a}^\top \left(\boldsymbol{p}^* - \boldsymbol{p}\right)\mathbb{I}\left(\boldsymbol{a}^\top \left(\boldsymbol{p}^* - \boldsymbol{p}\right) > 0\right)\right] - \mathbb{E}\left[\boldsymbol{a}^\top \left(\boldsymbol{p}^* - \boldsymbol{p}\right)\mathbb{I}\left(\boldsymbol{a}^\top \left(\boldsymbol{p}^* - \boldsymbol{p}\right) < 0\right)\right] \\ =&\mathbb{E}\left[\boldsymbol{a}^\top (\boldsymbol{p}^* - \boldsymbol{p})\left(\mathbb{I}\left(\boldsymbol{a}^\top (\boldsymbol{p}^* - \boldsymbol{p}\right) > 0) - \mathbb{I}\left(\boldsymbol{a}^\top (\boldsymbol{p}^* - \boldsymbol{p}) < 0\right)\right)\right] \\ \leq&\mathbb{E}\left[\left|\boldsymbol{a}^\top (\boldsymbol{p}^* - \boldsymbol{p})\right|\right] \\ \leq&\sqrt{\mathbb{E}\left[\left|\boldsymbol{a}^\top (\boldsymbol{p}^* - \boldsymbol{p})\right|^2\right]} \qquad \text{by Cauchy-Schwarz inequality} \\ =&\sqrt{\mathbb{E}\left[\boldsymbol{a}^\top (\boldsymbol{p}^* - \boldsymbol{p})(\boldsymbol{p}^* - \boldsymbol{p})^\top \boldsymbol{a}\right]} \\ \leq&\sqrt{\bar{a}^2 I \mathbb{E}\left[\|\boldsymbol{p}^* - \boldsymbol{p}\|_2^2\right]} \\ =&\bar{a}\sqrt{I}\mathbb{E}\left[\|\boldsymbol{p}^* - \boldsymbol{p}\|_2\right]\end{aligned}$$

□

B.4. *Proof of Theorem 1*

Proof. Recall that the sufficient condition for Theorem 1 is: only with the two events shown in Proposition 2 are satisfied, i.e. $\mathcal{E}_1 \cap \mathcal{E}_2$, we can assume the following inequality holds:

$$\|\boldsymbol{p}^* - \boldsymbol{p}\|_2 \leq \delta\epsilon \tag{B1}$$

where δ is a positive constant. The above inequality indicates that the L_2 distance between $\boldsymbol{p}^*$ and $\boldsymbol{p}$ is bounded by $\delta\epsilon$ (with high probability). As we know

$$\begin{aligned}
\mathbb{P}\left(\mathcal{E}_1 \cap \mathcal{E}_2\right) &\geq \mathbb{P}(\mathcal{E}_1) + \mathbb{P}(\mathcal{E}_2) - 1 \\
&= 1 - (1 - \mathbb{P}(\mathcal{E}_1)) - (1 - \mathbb{P}(\mathcal{E}_2)) \\
&= 1 - \mathbb{P}(\mathcal{E}_1^C) - \mathbb{P}(\mathcal{E}_2^C) \\
&\geq 1 - 2I \exp\left(-\frac{2T\epsilon^2}{I \max\limits_{i\in[I]}(\bar{R}_i - \underline{R}_i + \bar{a}_i)^2}\right) - 2I \exp\left(-\frac{2T\epsilon^2}{I \max\limits_{i\in[I]}(\bar{R}_i - \underline{R}_i)^2}\right)
\end{aligned}$$

the probability bound holds for all $\epsilon > 0$. Now, we have established a connection among probability events and the L_2 distance between $\boldsymbol{p}^*$ and $\boldsymbol{p}$ by using ϵ. If we further assume $\boldsymbol{p}_t^*$ is the optimal dual price vector within t timesteps, the probability of the L_2 distance between $\boldsymbol{p}^*$ and $\boldsymbol{p}_t^*$ exceeds $\delta\epsilon$ is

$$\begin{aligned}
\mathbb{P}\left(\|\boldsymbol{p}^* - \boldsymbol{p}_t^*\|_2 > \delta\epsilon\right) &= \mathbb{P}\left(\|\boldsymbol{p}^* - \boldsymbol{p}_t^*\|_2^2 > \delta^2\epsilon^2\right) \\
&\leq 2I \exp\left(-\frac{2t\epsilon^2}{I \max\limits_{i\in[I]}(\bar{R}_i - \underline{R}_i + \bar{a}_i)^2}\right) + 2I \exp\left(-\frac{2t\epsilon^2}{I \max\limits_{i\in[I]}(\bar{R}_i - \underline{R}_i)^2}\right)
\end{aligned}$$

With this probability bound, we can upper bound the L_2 distance between $\boldsymbol{p}^*$ and $\boldsymbol{p}_t^*$ by integration. Specifically, given that $\|\boldsymbol{p}^* - \boldsymbol{p}_t^*\|_2 \leq \frac{\bar{r}}{\underline{d}}$, we replace ϵ^2 with ϵ'

$$\begin{aligned}
&\mathbb{E}\left[\|\boldsymbol{p}^* - \boldsymbol{p}_t^*\|_2^2\right] \\
&= \int_0^{\frac{\bar{r}^2}{\underline{d}^2}} \mathbb{P}\left(\|\boldsymbol{p}^* - \boldsymbol{p}_n^*\|_2^2 > \delta^2\epsilon'\right) \mathrm{d}\epsilon' \\
&\leq \int_0^{\frac{\bar{r}^2}{\underline{d}^2}} \min\left\{2I \exp\left(-\frac{2t\epsilon'}{I \max\limits_{i\in[I]}(\bar{R}_i - \underline{R}_i + \bar{a}_i)^2}\right) + 2I \exp\left(-\frac{2t\epsilon'}{I \max\limits_{i\in[I]}(\bar{R}_i - \underline{R}_i)^2}\right), 1\right\} \mathrm{d}\epsilon' \\
&\leq \int_0^{\frac{\bar{r}^2}{\underline{d}^2}} \min\left\{4I \exp\left(-\frac{2t\epsilon'}{I \max\limits_{i\in[I]}(\bar{R}_i - \underline{R}_i + \bar{a}_i)^2}\right), 1\right\} \mathrm{d}\epsilon'
\end{aligned}$$

Next, we analyze the integral. With $\epsilon' = \frac{I \log I}{T} \cdot \varepsilon$, and let $\max_{i \in [I]} (\bar{R}_i - \underline{R}_i + \bar{a}_i)^2 = M$, we have

$$
\begin{aligned}
&\int_0^{\frac{\bar{r}^2}{d^2}} \min\left\{4I \exp\left(-\frac{2t\epsilon'}{IM}\right), 1\right\} \, \mathrm{d}\epsilon' \\
\leq &\frac{I \log I}{t} \int_0^{\infty} \min\left\{4I \exp\left(-\frac{2\varepsilon}{M} \log I\right), 1\right\} \, \mathrm{d}\varepsilon \\
\leq &\frac{I \log I}{t} \int_0^{\infty} \min\left\{4 \exp\left(\log I - \frac{2\varepsilon}{M} \log I\right), 1\right\} \, \mathrm{d}\varepsilon \\
= &\frac{I \log I}{t} \left[\int_0^{1.5M} \min\left\{4 \exp\left(\log I - \frac{2\varepsilon}{M} \log I\right), 1\right\} \, \mathrm{d}\varepsilon + \int_{1.5M}^{\infty} \min\left\{4 \exp\left(\log I - \frac{2\varepsilon}{M} \log I\right), 1\right\} \, \mathrm{d}\varepsilon\right] \\
\leq &\frac{I \log I}{t} \left[\int_0^{1.5M} 1 \, \mathrm{d}\varepsilon + \int_{1.5M}^{\infty} 4 \exp\left(\log I - \frac{2\varepsilon}{M} \log I\right) \, \mathrm{d}\varepsilon\right] \\
\leq &\frac{I \log I}{t} \left[1.5M + 2M \int_3^{\infty} \exp\left(\log I - x \log I\right) \, \mathrm{d}x\right] \\
= &M \frac{I \log I}{t} \left[1.5 + 2I \int_3^{\infty} I^{-x} \, \mathrm{d}x\right] \\
= &M \frac{I \log I}{t} \left[1.5 + \frac{2}{I^2 \log I}\right] \\
= &\frac{1}{t} \left[1.5MI \log I + \frac{2M}{I}\right]
\end{aligned}
$$

Therefore, we have

$$
\mathbb{E}\left[\|\boldsymbol{p}^* - \boldsymbol{p}_t^*\|_2\right] \leq \frac{N}{\sqrt{t}}
$$

where $N = \sqrt{1.5MI \log I + 2M/I}$ is a constant dependent only on $M = \max_{i \in [I]} (\bar{R}_i - \underline{R}_i + \bar{a}_i)^2$ and I □

B.5. *Inequality used in the proof of Theorem 1*

Let X be a continuous non-negative random variable. Prove that

$$
\mathbb{E}(X) = \int_0^{\infty} \mathbb{P}(X > 0) \mathrm{d}x
$$

Proof. For a continuous random variable:

$$
\mathbb{P}(X > x) = \int_x^{\infty} f_X(y) \mathrm{d}y
$$

where $f_X(\cdot)$ is the probability distribution function of variable X. Take the definite integral for $\mathbb{P}(X > x)$:

$$
\int_0^{\infty} \mathbb{P}(X > x) \mathrm{d}x = \int_0^{\infty} \int_x^{\infty} f_X(y) \mathrm{d}y \, \mathrm{d}x
$$

Then, we swap the order of integration since $f_X(\cdot)$ is strictly non-negative.

$$\begin{aligned}\int_0^\infty \mathbb{P}(X > x)\mathrm{d}x &= \int_0^\infty \int_0^y f_X(y)\mathrm{d}x \ \mathrm{d}y \\ &= \int_0^\infty f_X(y) \left(\int_0^y 1\mathrm{d}x \right) \ \mathrm{d}y \\ &= \int_0^\infty y f_X(y) \ \mathrm{d}y \\ &= \mathbb{E}(X)\end{aligned}$$

□

B.6. *Proof of Theorem 2*

Proof. Before we present the proof for the *Regret* of the DPOL algorithm, we first define two index sets for $i \in [I]$:

$$\begin{aligned}K_N &:= \{i : c_i > \mathbb{E}_{(r,\boldsymbol{a})\sim\mathcal{P},\boldsymbol{R}\sim\mathcal{Q}}[a_i \cdot \mathbb{I}(r > \boldsymbol{a}^\top \boldsymbol{p}^*) - R_{it}]\} \\ K_B &:= \{i : c_i = \mathbb{E}_{(r,\boldsymbol{a})\sim\mathcal{P},\boldsymbol{R}\sim\mathcal{Q}}[a_i \cdot \mathbb{I}(r > \boldsymbol{a}^\top \boldsymbol{p}^*) - R_{it}]\}\end{aligned}$$

where the subscripts "B" and "N" are short for binding and nonbinding, respectively. Assumption implies $K_B \cap K_N = \emptyset$ and $K_B \cup K_N = [I]$. By complementary slackness, type $i \in K_N$ service means the service inventory is enough for long-term consumption, where the dual price p_i is 0. However, $i \in K_B$ means the service inventory is possibly in long-term shortage, where the dual price p_i is larger than 0.

Next, for any dual-based online policy, its expected revenue

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=1}^T r_t y_t\right] \\ =&\mathbb{E}\left[\sum_{t=1}^T r_t y_t + \boldsymbol{l}_T^\top \boldsymbol{p}^*\right] - \mathbb{E}\left[\boldsymbol{l}_T^\top \boldsymbol{p}^*\right] \\ =&\mathbb{E}\left[\sum_{t=1}^T r_t y_t + \left(\boldsymbol{C} + \sum_{t=1}^T \boldsymbol{R}_t - \sum_{t=1}^T \boldsymbol{a}_{j(t)} y_t\right)^\top \boldsymbol{p}^*\right] - \mathbb{E}\left[\boldsymbol{l}_T^\top \boldsymbol{p}^*\right] \qquad \text{by the definition of } \boldsymbol{R}_t \\ =&\mathbb{E}\left[\sum_{t=1}^T \left(r_t y_t + \boldsymbol{c}^\top \boldsymbol{p}^* + \boldsymbol{R}_t^\top \boldsymbol{p}^* - \boldsymbol{a}_{j(t)}^\top \boldsymbol{p}^* y_t\right)\right] - \mathbb{E}\left[\boldsymbol{l}_T^\top \boldsymbol{p}^*\right] \\ =&\sum_{t=1}^T \mathbb{E}\left[r_t y_t + \boldsymbol{c}^\top \boldsymbol{p}^* + \boldsymbol{R}_t^\top \boldsymbol{p}^* - \boldsymbol{a}_{j(t)}^\top \boldsymbol{p}^* y_t\right] - \mathbb{E}\left[\boldsymbol{l}_T^\top \boldsymbol{p}^*\right] \qquad \text{exchange summation and expectation} \\ \geq&\sum_{t=1}^T \mathbb{E}\left[\mathbb{E}\left[\left(r_t y_t + \boldsymbol{c}^\top \boldsymbol{p}^* + \boldsymbol{R}_t^\top \boldsymbol{p}^* - \boldsymbol{a}_{j(t)}^\top \boldsymbol{p}^* y_t\right) \mid \mathcal{H}_{t-1}\right]\right] - \mathbb{E}\left[\boldsymbol{l}_T^\top \boldsymbol{p}^*\right] \text{ place a conditional expectation} \\ =&\sum_{t=1}^T \mathbb{E}\left[g\left(\boldsymbol{p}_t\right)\right] - \frac{\bar{r}}{\underline{d}} \mathbb{E}\left[\sum_{t=1}^T \sum_{i\in K_B} c_i + R_{it} - a_{ij(t)}\right]\end{aligned}$$

where at each timestep t, a vector $\boldsymbol{p}_t$ is computed based on historical data $\boldsymbol{p}_t = h_t(\mathcal{H}_{t-1})$,

$\mathcal{H}_{t-1} = \{r_\tau, \boldsymbol{a}_{j(\tau)}, \boldsymbol{R}_\tau, y_\tau\}_{\tau=1}^{t-1}$. Note that service indices $i \in K_B$ appear at the bottom of the above inequalities. Services $i \in K_N$ are not included in calculating the expected revenue because their corresponding dual prices p_i are all 0 to be omitted.

Therefore, the *Regret* is upper-bounded by

$$
\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{T} r_t y_t^*\right] - \mathbb{E}\left[\sum_{t=1}^{T} r_t y_t\right] \\
\leq &\sum_{t=1}^{T} \mathbb{E}\left[g\left(\boldsymbol{p}^*\right) - g\left(\boldsymbol{p}_t\right)\right] + \frac{\bar{r}}{\underline{d}} \mathbb{E}\left[\sum_{t=1}^{T} \sum_{i \in K_B} c_i + R_{it} - a_{ij(t)} \mathbb{I}\left(r_t > \boldsymbol{a}_t^\top \boldsymbol{p}^*\right)\right] \\
\leq &\sum_{t=1}^{T} \bar{a} N \sqrt{\frac{I}{t}} + \frac{\bar{r}}{\underline{d}} \sum_{i \in K_B} \mathbb{E}\left[\sum_{t=1}^{T} c_i + R_{it} - a_{ij(t)} \mathbb{I}\left(r_t > \boldsymbol{a}_t^\top \boldsymbol{p}^*\right)\right] \\
\leq &2\bar{a} N \sqrt{IT} + \frac{\bar{r}}{\underline{d}} \sum_{i \in K_B} \sqrt{\mathbb{E}\left[\left(\sum_{t=1}^{T} c_i + R_{it} - a_{ij(t)} \mathbb{I}\left(r_t > \boldsymbol{a}_t^\top \boldsymbol{p}^*\right)\right)^2\right]} \\
= &2\bar{a} N \sqrt{IT} + \frac{\bar{r}}{\underline{d}} \sum_{i \in K_B} \sqrt{\mathbb{E}\left[\left(\sum_{t=1}^{T} \mathbb{E}\left[a_{ij(t)} \mathbb{I}\left(r_t > \boldsymbol{a}_t^\top \boldsymbol{p}^*\right)\right] - a_{ij(t)} \mathbb{I}\left(r_t > \boldsymbol{a}_t^\top \boldsymbol{p}^*\right)\right)^2\right]} \\
= &2\bar{a} N \sqrt{IT} + \frac{\bar{r}}{\underline{d}} \sum_{i \in K_B} \sqrt{\sum_{t=1}^{T} \operatorname{Var}\left[a_{ij(t)} \mathbb{I}\left(r_t > \boldsymbol{a}_t^\top \boldsymbol{p}^*\right)\right]} \\
= &2\bar{a} N \sqrt{IT} + \frac{\bar{r}\bar{a}|K_B|}{\underline{d}} \sqrt{T} \\
= &\left(2\bar{a} N \sqrt{I} + \frac{\bar{r}\bar{a}|K_B|}{\underline{d}}\right) \sqrt{T}
\end{aligned}
$$

N is a constant appeared at the end of Appendix B.4. □