APPLICATIONES MATHEMATICAE
44,2 (2017), pp. 215–245

V. S. Borkar (Mumbai)
K. Ravikumar (Cincinnati, OH)
Krishnakant Saboo (Mumbai and Urbana, IL)

AN INDEX POLICY FOR DYNAMIC PRICING IN CLOUD COMPUTING UNDER PRICE COMMITMENTS

Abstract. A dynamic pricing based resource allocation problem for cloud computing is cast as a Markov decision process with average reward and hard per time combinatorial constraints. Following Whittle, its relaxation as a constrained average reward Markov decision process is analyzed and its Whittle indexability is established. An iterative scheme to compute the Whittle indices is also proposed.

1. Introduction. *Cloud computing* ([1]) is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. The emergence of cloud computing has brought in a novel trend of purchasing and consuming Information Technology (IT) services on demand. The technology has recently garnered a lot of traction in industry because in many respects the cloud resembles a utility that supplies water or electric power: with the cloud, users can access the IT resources at any time and from multiple locations, track their usage levels, and scale up their service delivery capacity as needed, without large upfront investments in software or hardware. Also, in contrast to the traditional distributed systems such as grids and clusters which mainly focused on improvement of the system performance in terms of response time and

2010 *Mathematics Subject Classification*: Primary 90B36; Secondary 68M20, 93E20, 60J20.
Key words and phrases: cloud computing, IaaS, constrained Markov decision process, Whittle index, dynamic programming.
Received 31 July 2016; revised 7 June 2017.
Published online 25 August 2017.

([1]) As defined by the National Institute of Standards and Technology (NIST).

DOI: 10.4064/am2313-6-2017

throughput, cloud computing has brought in monetary dimension to service delivery and paved way for the following business models based on the stack structure of cloud architecture depicted in Figure 1:

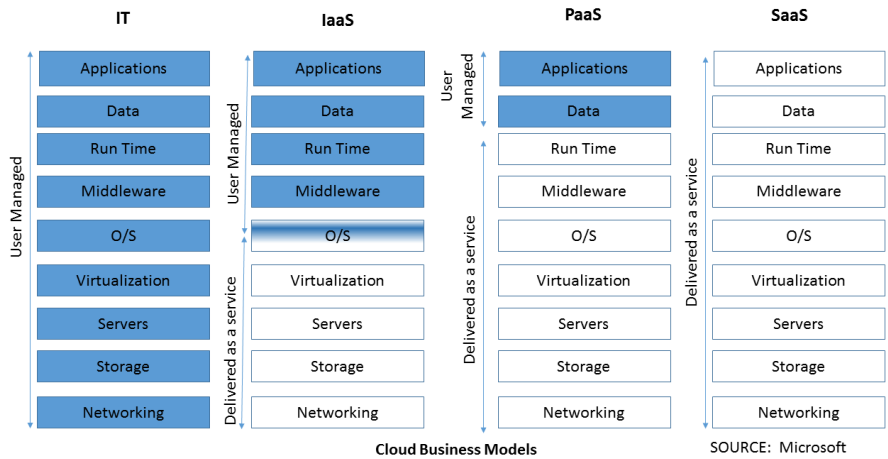

Fig. 1. IT versus cloud business models

- **Infrastructure-as-a-Service (IaaS)** is a self-service model for accessing, monitoring, and managing remote data center infrastructures, such as compute (virtualized or bare metal), storage, networking, and networking services (e.g. firewalls). Instead of having to purchase hardware outright, users can purchase IaaS based on consumption, similar to electricity or other utility billing. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, and deployed applications. Examples include Amazon Web Services (AWS), Cisco Metapod, Microsoft Azure, and Google Compute Engine (GCE).
- **Platform-as-a-Service (PaaS)** provides businesses with an independently maintained platform upon which their web applications can be built, refined and deployed. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly configuration settings for the application-hosting environment (e.g., Microsoft Azure, Salesforce Heroku, AWS Elastic Beanstalk).
- **Software-as-a-Service (SaaS)** enables the consumer to use the providers applications running on a cloud infrastructure. The applications are accessible from various client devices through either a thin client interface, such as a web browser (e.g., web-based email), or a program interface (API). The consumer does not manage or control the underlying cloud

infrastructure or even individual application capabilities, with the possible exception of limited user-specific application configuration settings (e.g., Webmail service).

As one would expect, estimating the cost of service provisioning, and hence pricing, becomes complex as one moves from IaaS to SaaS because workloads become more and more complex and differentiated at SaaS level. Interestingly, in the case of IaaS, virtualization technology allows cloud providers to run multiple so-called virtual machines (VMs) on one physical machine. Each VM instance can run independently of others and can be tracked for its usage. A VM is transparent to the application and its end-user and further has the benefit that each machine can be custom-tailored towards the needs of the users with respect to the technical requirements and pre-installed software libraries. VM instances of varying sizes—such as small, medium or large instances which differ in CPUs, memory, and storage—are generated on demand from the physical server and are then allocated or *provisioned* to users. Any active instance can be deprovisioned or released at any time or upon the completion of assigned job after which the virtual instance vanishes. From the providers point of view, virtualization enables efficient utilization of physical machines. While it is possible to generate as many virtual instances as desired, physical server capacity, computing performance requirement, and the cost to configure new instances at run time, may limit the number of instances that can be hosted at any time on the physical server.

The most frequently used pricing model of the IaaS providers is *pay-per-use*, in which the user pays a static price for a used VM instance. The pay-per-use pricing model is a simple model, in which units (or units per time) are associated with fixed price values. Differential pricing is implemented by way of creating VM instances of different types and configurations that primarily differ in their resource consumption. A similar but different pricing model is *subscription*, where the user subscribes (signs a contract) for using a pre-selected combination of service units for a fixed price and longer time frame, usually monthly or yearly. The dominance of the above pricing models is due to the fact that users often prefer simple pricing models (like pay-per-use or subscription) with a static payment fee for ease of understanding and accounting.

While static pricing is dominant today, dynamic pricing is emerging as an attractive alternative to cope with unused capacities and uncertain demand patterns. It is important to observe that computing resources such as CPU and network bandwidth are inherently perishable: unused capacity is a lost opportunity to the provider. Revenue management in cloud is in this sense similar in spirit to management practices in airlines/hotel/car rentals but

with a difference. Unlike in the airlines scenario, the exact usage duration of an instance is not known *a priori*.

Amazon EC2 introduced auction mechanism for dynamic pricing of spot instances to derive revenues from unused capacity. But under the auction driven spot pricing, users are inconvenienced through preemption leading to undesirable delays in workload execution. As a result, participation in spot auctions has noticeably decreased over time [2]. Motivated by this fact, we work within the realm of dynamic pricing but instead of using a market mechanism, we allow the cloud provider to reset prices (more precisely, price per unit time) by weighing instantaneous demand against available capacity and the associated cost of service provisioning. This price-setting power helps the provider to reclaim capacity when in need by making price unattractive to arriving customers. We also assume that the service provider makes a *price and service commitment* to an arriving customer; that is, a job receives uninterrupted service (with no preemption) and the *price rate* charged on the job remains constant until its completion and is set equal to the price rate at which it joined the system (see Figure 2). In other words, price commitment is a desirable feature for business customers, and it also retains the *ease of accounting* feature of static pricing (but may entail elaborate book-keeping by the provider—which is cut short by the modelling artifice presented in the ensuing sections). Also, since all the customers arriving at any given time see the same price, the suggested pricing is non-discriminatory. We analyze dynamic pricing under the above assumptions.

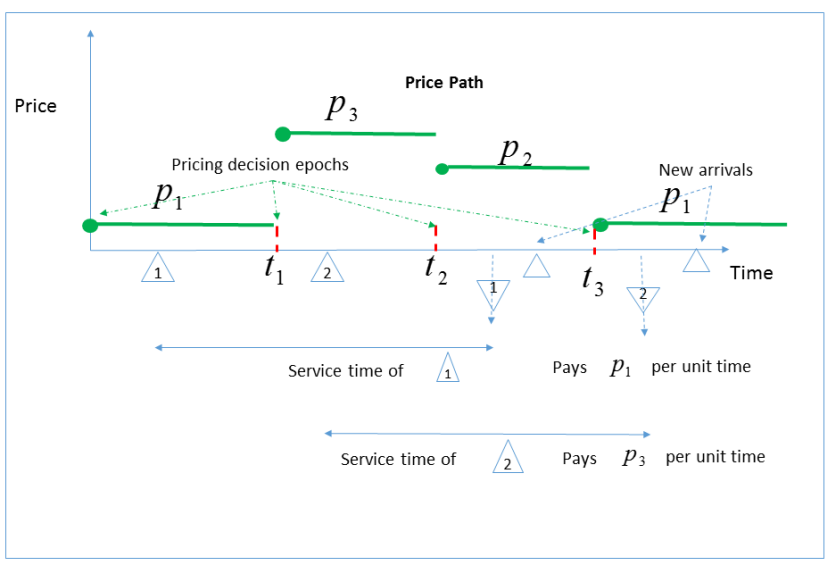

Fig. 2. A sample path under price commitments

It is important to note that the dynamic pricing described here is different from the self-selection of prices by customers which is based on their differential tolerance to waiting, as is observed in services offered with priority and normal queues. It resembles closely the *time of the day* toll/pricing practiced in utility delivery. But, unlike utility services, in cloud delivery it is hard to extract fixed timings of peak usage or normal usage in a day. VM usage fluctuation will depend on the usage pattern of the application hosted on the VM instance. It may be appropriate to classify the dynamic pricing described here as *load or state dependent pricing.*

In the sections below, we develop a model for the above setting and derive an index policy using the restless bandit framework.

2. The model. In the above dynamic price setting, implementing price commitment will entail keeping track of joining and departure times of each job (or starting and stopping the associated Virtual Machine Instance) and the price rate at which it joins. To simplify the book-keeping, we assume that the cloud provider operates within a finite set of prices. Without loss of generality, we assume that these prices are within the range of the cost rate and the existing market price rate of virtual instances. We associate with each price a virtual queue, arrivals into which will happen only when the operating price is set equal to the price corresponding to that queue. In other words, as many virtual queues as the number of prices are created and at any time only one virtual queue sees arrivals and the rest see only departures. Every arrival is associated with a virtual machine instance and it is assumed that the physical server is capable of creating as many virtual instances as desired. But cost consideration may limit the number. As each job is necessarily associated with a virtual queue, its joining price is tagged, and hence is easily tracked. It is important to note that there are no physical queues for resource contention but only virtual queues or *job clusters*—clustered by *price* in an *infinite server* queue. Refer to Figure 3 for a schematic diagram of the system.

Now, in the above set up, consider the cloud computing facility with the objective of deciding the price at which a given computing resource should be allocated. We model the computing facility as infinitely many identical machines partitioned into N clusters corresponding to N virtual queues of Figure 3. Each cluster corresponds to a different price p_i, $1 \leq i \leq N$. Without loss of generality we assume that p_i strictly increases with i:

$$p_1 < \cdots < p_N.$$

At price p_i, jobs arrive independently with identical Poisson distribution with mean Λ_i which decreases with increasing i:

$$\Lambda_1 > \cdots > \Lambda_N.$$

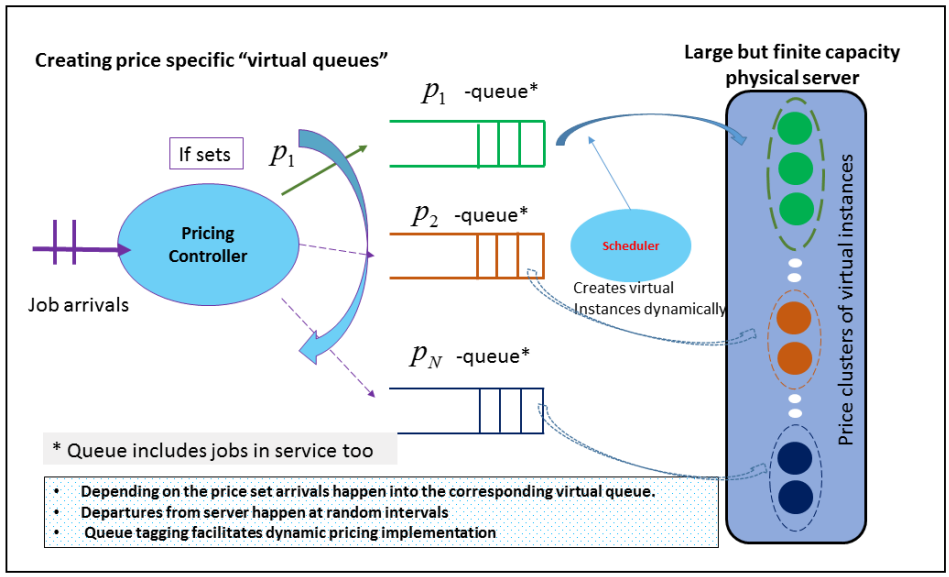

Fig. 3. System model with finite prices

Only one price is offered at a given time, hence only one cluster admits new arrivals, the one corresponding to the stated price. We shall say that this cluster is 'active' and the other clusters are 'passive'. Jobs come with predefined price thresholds of their own and join the active cluster if the price being offered by the system is less than or equal to their price threshold. This is accounted for in the aforementioned monotone dependence of the arrival rate on price. Since the clusters are infinite, there is zero waiting time, i.e., as soon as a job arrives into the system it is allotted an empty server—there is no queuing. At every instant, the probability of the job leaving the server is q. Hence, the departure process is Bernoulli with distribution depending on the number of jobs being served in the cluster at that point of time. In particular, there can be jobs being served in passive queues, and hence there can be departures from them, but no arrivals. Let the number of jobs in cluster i (i.e., at price p_i) at time n be X_n^i. The dynamics for cluster i can then be written as: for $1 \leq i \leq N$,

$$X_{n+1}^i = X_n^i + \nu_n^i \xi_{n+1}^i - D_{n+1}^i \tag{1}$$

where

- i is the cluster of price p_i such that p_i increases with i;
- $\xi_{n+1}^i \sim$ i.i.d. Poisson with rate Λ_i that decreases with i;
- $\nu_n^i = 1$ if cluster i is active, $= 0$ otherwise;

- $\{D_n^i\}$ is the departure process with conditionally Bernoulli distribution:

$$P(D_{n+1}^i = k \mid X_n^i = m) = \begin{cases} 0 & \text{if } k > m, \\ \binom{m}{k} q^k (1-q)^{m-k} & \text{if } 0 \le k \le m. \end{cases}$$

Our aim will be to maximize the long run expected reward:

$$\max \varlimsup_{T\to\infty} \frac{1}{T} \sum_{n=0}^{T-1} \sum_{i=1}^{N} p_i (E[X_n^i] - \nu_n^i k_i) \tag{2}$$

subject to

$$\sum_i \nu_n^i = 1, \tag{3}$$

$$\sum_i X_n^i \le M. \tag{4}$$

Here k_i is a cost that reflects lost clients due to non-affordability of the stated price for a section of clients. The optimization is over non-anticipative choices of the 'control' variables $\{\nu_n^i\}$, i.e., $\nu_n := [\nu_n^1, \dots, \nu_n^N]$ is allowed to depend on past and present observations X_m^i, $m \le n$, and independent extraneous randomization, but not on future values X_m^i, $m > n$. The last constraint (4) is a hard constraint on the total number of machines that can be active at any given time. As there is no queuing, each server is either idle or has one job in service with no jobs waiting. We shall relax this hard constraint to an average constraint, i.e., we replace the above problem by

$$\max \varlimsup_{T\to\infty} \frac{1}{T} \sum_{n=0}^{T-1} \sum_{i=1}^{N} ((p_i - \mu) E[X_n^i] - p_i \nu_n^i k_i) \tag{5}$$

subject to

$$\sum_i \nu_n^i = 1, \tag{6}$$

where the constraint on the total number of active machines has been absorbed as a negative reward term $-\mu X_n^i$. We can interpret μ as the associated Lagrange multiplier or penalty. We assume this parameter is known, or equivalently pre-selected ([2]). The last remaining constraint (3) is again a hard constraint imposing the restriction that only a single price be offered at a time. This will need a further relaxation to an average constraint along similar lines to the above. We introduce it after briefly recalling the theory of Whittle indexability in the next section. Before doing so, we make some observations regarding the constants above, which also put some natural constraints on them.

([2]) It is possible to extend our scheme to one where even μ is iteratively learnt, by using a primal-dual framework. We touch upon this in the concluding section.

1. We may assume that $p_i - \mu > 0$ for all i. If $p_i - \mu < 0$ for all i, it is optimal to never admit a job, i.e., $\nu^i(n) \equiv 0$ for all i, n. If it is so for some i, those choices may be dropped from consideration altogether as not being viable. Thus there is no loss of generality in assuming that $p_i - \mu > 0$ for all i.
2. Given that $p_i - \mu > 0$ for all i, if $\Lambda_1 \geq q$, the choice $\nu^1(n) \equiv 1$ for all n leads to the stationary expectation of X_n^1 being $+\infty$, thus leading to infinite reward. Thus it makes sense to assume that $q > \Lambda_1$, implying $q > \Lambda_i$ for all i.
3. We expect k_i to be proportional to $\Lambda_1 - \Lambda_i$, though this fact will not be needed for our analysis.

One immediate consequence of $q > \max_i \Lambda_i$ is that for the combined process $X_n := [X_n^1, \ldots, X_n^N]$, the function $\Phi([x_1, \ldots, x_N]) := \sum_i x_i$ serves as a stochastic Lyapunov function: for

$$\mathcal{F}_n := \sigma(X_m^i, \nu_m^i : 1 \leq i \leq N,\ 0 \leq m \leq n),\ n \geq 0, \text{ and } \theta := [0, \ldots, 0] \in \mathbb{R}^N,$$

we have

$$E[\Phi(X_{n+1}) \mid \mathcal{F}_n] - \Phi(X_n) \leq -q\Phi(X_n) + \Lambda_1 < 0 \quad \text{when } X_n \neq \theta, \tag{7}$$

regardless of the choice of $\{\nu_n^i\}$. We shall be specifically interested in the stationary policies, that is, $\bar{\nu}_n = [\nu_n^1, \ldots, \nu_n^N]$ of the form $\nu_n^i = v^i(X_n)$ for all i, n for some $v^i : \mathbb{N} := \{0, 1, 2, \ldots\} \to \{0, 1\}$ and $1 \leq i \leq N$. Under such policies, $\{X_n\}$ is a time-homogeneous Markov chain. By standard abuse of notation, we shall identify such a policy with the map $v(\cdot) := [v^1(\cdot), \ldots, v^N(\cdot)] : \mathbb{N}^N \mapsto$ the unit coordinate vectors in $\mathbb{R}^N$. Let $\mathcal{P}(\mathbb{N}) :=$ the space of probability measures on $\mathbb{N}$ with Prokhorov topology. From (7), we then have the following:

LEMMA 2.1. *Under any stationary policy, $\{X_n\}$ has a single aperiodic communicating class that includes θ (possibly with some transient states), and restricted to this communicating class, the chain is geometrically ergodic. Furthermore, if π denotes its unique stationary distribution and $\tau_\theta := \{n \geq 0 : X_n = \theta\}$, the first hitting time of θ, then:*

1. *For every $x \neq \theta$, there exist $a > 0$ such that $E[e^{a\tau_\theta} \mid X_0 = x] \leq K_x < \infty$ for some $K_x < \infty$, uniformly in v.*
2. *For every $f : \mathbb{N} \to \mathbb{R}$ that is $O(\Phi(\cdot))$, $E[f(X_n)] \to \sum_x \pi(x) f(x)$ exponentially, uniformly in v, in fact, for $x = [x_1, x_2, \ldots]$,*

$$\left| E[f(X_n) \mid X_0 = x] - \sum_y \pi(y) f(y) \right| \leq K(1 + |x|)\eta^n \tag{8}$$

for some $K > 0$ and $0 < \eta < 1$ with $|x| := \sum_i x_i$.

3. *$\sup E[\sum_i X_n^i] < \infty$, where the supremum is over all stationary policies and the expectation is over the corresponding stationary distributions. In particular, the latter are compact in $\mathcal{P}(\mathbb{N})$.*

4. $\sup E[\sum_{m=0}^{\tau_\theta} X_m] < \infty$, *where the supremum is over all admissible controls.*

Proof. Fix v. Aperiodicity follows from the observation that there is a non-zero probability of remaining in state θ. In view of (7), it follows from [10, Theorem 16.0.1, p. 393] that $\{X_n\}$ is in fact Φ-uniformly ergodic in the sense of [10, (16.2), p. 392], in particular, uniformly ergodic in the sense of [10, (16.6), p. 393] and geometrically ergodic. The main claim and the first bullet follow by specializing [10, Theorem 16.0.2, p. 394] to the present case, the second follows likewise from [10, Theorem 16.0.1, p. 393]. The uniformity in v follows in view of the common Lyapunov condition (7). The last two bullets follow as in [5, Theorem 8.1, p. 108]. ∎

3. Whittle index: an introduction. We now briefly summarize the set-up of Whittle indexability. Consider N Markov chains $\{X_n^i\}$, $1 \le i \le N$, on discrete state spaces S^i, $1 \le i \le N$, resp. Each chain has two possible modes, active and passive, with corresponding transition probabilities $p_a^i(j|k), p_b^i(j|k)$ resp. for $j,k \in S^i$, $1 \le i \le N$, and per stage rewards $r_a^i, r_b^i : S^i \to \mathbb{R}$ resp. with $r_a^i(\cdot) \ge r_b^i(\cdot)$. (This formulation is slightly more general than the original Whittle formulation in that we allow for a low but non-zero reward for passivity. This does not change the analysis in any essential manner.) At most M out of N chains, $1 \le M < N$, are allowed to be active. Let

$$\nu_n^i := I\{\text{the } i\text{th chain is active at time } n\}.$$

Then the objective is to maximize

$$\limsup_{n\uparrow\infty} \frac{1}{n} \sum_{m=0}^{n-1} \sum_i E[\nu_m^i r_a^i(X_m^i) + (1-\nu_m^i) r_b^i(X_m^i)]$$

subject to

$$\sum_i \nu_n^i \le M \quad \forall n, \tag{9}$$

over non-anticipative choices of $\{\nu_n^i\}$. The hard per stage constraint (9) makes this problem difficult [13], more so than the case when passive chains remain frozen. The latter is the classical multiarmed bandit scenario for which an elegant optimal index policy is available in the form of Gittins index [7]. Here, however, the chains have a 'neutral' dynamics governed by $\{p_b^i(\cdot|\cdot)\}$ even when they are passive, making them 'restless' bandits. Finding the exact optimal policy turns out to be a hard problem, so Whittle introduced a relaxation wherein the per stage constraint (9) is replaced by

an average constraint

$$\limsup_{n\uparrow\infty} \frac{1}{n} \sum_{m=0}^{n-1} E\Big[\sum_i \nu_m^i\Big] \le M \quad \forall n. \tag{10}$$

The decision variables or controls are the $\{\nu_m^i\}$ which have to be chosen non-anticipatively. That is, the choice of which bandits to activate has to be based on past and present observations and past decisions, and possibly some extraneous independent randomization, but not on future. This is then a constrained Markov decision process with state space $\prod_{i=1}^N S^i$, separable (average) cost and separable (average cost) constraints. While this can be approached using techniques from [1], the problem remains hard, particularly in view of the exponential blow-up of the state space with N [13]. This motivated Whittle to formulate a heuristic index policy based on a decomposition of the problem into N individual problems on state spaces S^i, $1 \le i \le N$. If $|S^i|$ is independent of i, then it is clear that the state space in the original problem grows exponentially with N, while it does so only linearly if the problem splits in a manner described above, thereby ensuring a major computational advantage.

The Whittle scheme is as follows. Consider the following individual problem for each i:

$$\max \limsup_{n\uparrow\infty} \frac{1}{n} \sum_{m=0}^{n-1} E[\nu_m^i r_a^i(X_m^i) + (1-\nu_m^i)(\lambda + r_b^i(X_m^i))],$$

where λ is the Whittle 'subsidy' for passivity. This formalism is derived from the Lagrange multiplier formulation of the constrained Markov decision process above with average constraint (10). The problem is said to be *Whittle indexable* if the set of states where it is optimal to be passive increases *monotonically* from empty set to the entire state space as λ increases from $-\infty$ to ∞. If so, for each state x, let $\lambda_i(x)$ denote the value of λ for which both active and passive behaviors are equally attractive. This is formally characterized as follows. The dynamic programming equation for the above Markov decision process is [14]

$$V(x) = \max\Big(r_a(x) + \sum_y p_a(y \mid x)V(y), \lambda + r_b(x) + \sum_y p_b(y \mid x)V(y)\Big) - \beta^*, \tag{11}$$

where V is the value function and $\beta^* :=$ the optimal reward. Then

$$\lambda_i(x) = r_a(x) - r_b(x) + \Big(\sum_y p_a(y \mid x)V(y) - \sum_y p_b(y \mid x)V(y)\Big). \tag{12}$$

If the overall state at time n is $X_n = x = [x_1, \dots, x_N]^T$, then order the $\{\lambda_i(x_i) : 1 \le i \le N\}$ in decreasing order, say

$$\lambda_{i_1}(x_{i_1}) \ge \cdots \ge \lambda_{i_N}(x_{i_N}),$$

breaking any tie arbitrarily. The Whittle heuristic then is to keep the top M of the indices $i_1, \dots, i_M$ active and the rest passive.

This scheme, albeit heuristic, has seen many successful applications. Some recent examples are [9], [11], [12], [15]. Asymptotic optimality has also been established in the 'infinite bandits' limit [19]. See [7], [8], [16] for an overview of index theory for bandits, restless or otherwise.

4. The dynamic programming equation. Returning to our original problem set-up, we analyze the control problem for individual chains. We fix a cluster i for the time being and drop the corresponding sub/superscript in order to simplify the notation. We can combine the dynamics (1) with the general dynamic programming equation (11) for the average cost problem in order to come up with the dynamic programming equation for the present case as follows:

$$\begin{aligned}(13)\qquad V(x) &= (p-\mu)x - \beta \\ &\quad + \max\Bigg(-k + \sum_{d=0}^{x} \binom{x}{d} q^d (1-q)^{x-d} \sum_{a=0}^{\infty} \frac{\Lambda^a(p)}{a!} e^{-\Lambda} V(x+a-d), \\ &\qquad\qquad \lambda + \sum_{d=0}^{x} \binom{x}{d} q^d (1-q)^{x-d} V(x-d)\Bigg).\end{aligned}$$

For notational simplicity, we shall write $c(x,\nu) := (p-\mu)x + (1-\nu)\lambda - \nu k$, $x \in \mathbb{N}$, $\nu \in \{0,1\}$, as the 'per stage reward' function. Also, denote by $p_a(y \,|\, x, \nu)$, $p_b(y \,|\, x, \nu)$ the controlled transition probabilities in the active, resp. passive mode featuring in (13). Recall that stationary policies are control policies of the form $\nu_n = v(X_n)$, $n \ge 0$, for some $v : \mathbb{N} \to \{0,1\}$, which we identify with the map v itself. Let $W(x) = x$, $x \in \mathbb{N}$.

LEMMA 4.1. *Under any admissible control sequence $\{\nu_n\}$, for $\mathcal{G}_n := \sigma(X_m, \nu_m, m \le n), n \ge 0$,*

$$E[W(X_{n+1}) \,|\, \mathcal{G}_n] - W(X_n) \le -qX_n + \Lambda_1 \qquad \forall n.$$

In particular, all stationary Markov policies $v(\cdot)$ are stable, and letting $E[\cdot], E_v[\cdot]$ denote the expectations under an admissible control, resp. a stationary Markov policy v, for $\tau_0 := \min\{n > 0 : X_n = 0\}$, we have:

1. *For every $x \ne \theta$, there exists $c > 0$ such that $E[e^{c\tau_0} \,|\, X_0 = x] \le K_x < \infty$ for some $K_x < \infty$, uniformly in v.*

2. *For every* $f : \mathbb{N} \to \mathbb{R}$ *that is* $O(|x|)$, $E_v[f(X_n)] \to \widetilde{E}_v[f(x)]$ *exponentially, uniformly in* v, $\widetilde{E}_v[\,\cdot\,]$ *being the expectation with respect to the corresponding stationary distributions.*
3. $\sup_v \widetilde{E}_v[\sum_i X_n^i] < \infty$. *In particular, the stationary distributions under stationary policies form a compact subset of* $\mathcal{P}(\mathbb{N})$.
4. $\sup E[\sum_{m=0}^{\tau_0} X_m] < \infty$, *where the supremum is over all admissible controls.*

This follows as in Lemma 2.1. Since state 0 is reachable under any stationary policy, this is a unichain problem in the sense of [14, p. 348].

LEMMA 4.2. *There exists an optimal stable stationary policy* $v^* : \mathbb{N} \to \{0, 1\}$ *and* $W(x) = x$ *serves as the Lyapunov function for the corresponding optimal Markov chain* $\{X_n^*\}$.

Proof. Since $p > \mu$, the function c above satisfies

$$\lim_{x \uparrow \infty} \max_{\nu} c(x, \nu) = \infty.$$

Thus it is 'near-monotone' in the sense of [3, (1.5), p. 58]. The first claim then follows by [3, Theorem 1.1, p. 58]. (This is proved under irreducibility assumption, but the same arguments work for the unichain case: see, e.g., [4, Lemmas 11.8 and 11.9, p. 353] for an even more general result.) The second claim follows from Lemma 2.1. ∎

THEOREM 4.1. *Equation* (13) *with the additional condition*

$$V(0) = 0 \tag{14}$$

has a unique solution $(V(\cdot), \beta)$ *where* $\beta :=$ *the optimal reward and*

$$V(i) = \min_v E\Big[\sum_{m=1}^{\tau_0} (c(X_m, v(X_m)) - \beta) \;\Big|\; X_0 = i\Big] \tag{15}$$

where the minimum is over all stationary policies $v(\cdot)$.

Proof (sketch). This follows as in [3, Chapter VI], the only difference being that we use the weaker hypothesis of unichain property rather than irreducibility. One important point to note is the following. Suppose for a stationary policy v, state i is in the single positive recurrent communicating class thereof and we change the corresponding control $v(i)$ to (say) $v'(i)$, leading to a positive probability of transition from i to some state $j \neq i$ for which this probability was zero under v. By the unichain property, there is a path from j to 0 and since under v there was a path from 0 to i, there is now a path from 0 to j, so j is in the single positive recurrent communicating class of the new policy. This replaces the 'stability under local perturbations' property defined in [3, p. 71]. The proof now proceeds in the following steps.

Let v^* be a stable optimal stationary policy guaranteed by Lemma 4.2 and $\{X_n^*\}$ the corresponding optimal process as in Lemma 4.2. Let π^* denote the corresponding stationary distribution and S^* its support. Define

$$V(i) = E\Big[\sum_{m=1}^{\tau_0}(c(X_m^*, v^*(X_m^*)) - \beta) \;\Big|\; X_0 = i\Big], \qquad i \in S^* \setminus \{0\}, \tag{16}$$

$$V(i) = \min_v E\Big[\sum_{m=0}^{\zeta-1}(c(X_m, \nu_m) - \beta) + V(\zeta) \;\Big|\; X_0 = i\Big], \quad i \notin S^*, \tag{17}$$

with $V(0) = 0$, where $\zeta := \min\{n \geq 0 : X_n \in S^*\}$ and the minimum in (17) is over all stationary policies v. Then:

Step 1. V defined above satisfies (13). This follows as in [3, Theorem 2.1, pp. 76–78] for S^* and by (17) for the rest.

Step 2. Among all solutions $(V'(\cdot), \beta')$ of (13) satisfying $V'(0) = 0$, the solution $(V(\cdot), \beta)$ is the unique one wherein V is $O(x)$. For S^* alone this follows as in [3, Theorem 4.2, pp. 89–90]. For $(S^*)^c$, V is the unique value function for the dynamic program with the cost

$$E\Big[\sum_{m=0}^{\zeta-1}(c(X_m, \nu_m) - \beta) + V(\zeta) \;\Big|\; X_0 = i\Big].$$

Step 3. The representation (15) holds. For states in S^*, this follows as in [3, Lemma 2.5, pp. 79–81]. For the rest it follows from (17) combined with the dynamic programming principle.

Step 4. Any choice of maximizers on the right hand side of (13) yields an optimal stable stationary policy. This follows as in [3, Theorem 3.2 and Corollary 3.3, p. 84] for states in the corresponding communicating class, and trivially for the rest.

This concludes the proof. ∎

For a 'discount factor' $\alpha \in (0, 1)$, define the infinite horizon discounted reward

$$J_\alpha(i, \{\nu_m\}) := E\Big[\sum_{m=0}^{\infty} \alpha^m c(X_m, \nu_m) \Big| X_0 = i\Big].$$

The corresponding value function is $V_\alpha(i) := \max J_\alpha(i, \{\nu_m\})$ where the maximum is over all admissible control policies and can be replaced by maximum over all stationary policies (see, e.g., [3, Chapter III]). Then V_α satisfies the dynamic programming equation

$$V_\alpha(i) = \max_u \Big[c(i, u) + \alpha \sum_j p(j \mid i, u) V_\alpha(j)\Big] \tag{18}$$

and any $v : \mathbb{N} \to \{0,1\}$ such that $v(i)$ is the argmax on the right, defines an optimal stationary policy (see *ibid.*). Then $\overline{V}_\alpha := V_\alpha(\cdot) - V_\alpha(0)$ satisfies

$$\overline{V}_\alpha(i) = \max_u \Big[c(i,u) - (1-\alpha)V_\alpha(0) + \alpha \sum_j p(j \mid i,u)\overline{V}_\alpha(j) \Big]. \tag{19}$$

Denote by $\hat{J}_\alpha(i,v)$ the α-discounted reward above for the stationary policy v and initial condition i. Then by a Tauberian theorem [18], $\lim_{\alpha\uparrow 1} \hat{J}_\alpha(i,v) = \beta(v) :=$ the stationary expectation of the reward under v.

LEMMA 4.3. $\lim_{\alpha\uparrow 1} \overline{V}_\alpha = V$.

Proof. Let v_α denote an optimal stationary policy and $\{X_n\}$ a chain controlled by v_α with initial condition depending on the context. Let β_α denote the corresponding stationary expectation of the reward. Then $\beta_\alpha \le \beta$. Also,

$$|E[c(X_n, v_\alpha(X_n))] - \beta_\alpha| \le K\eta^n \tag{20}$$

for some $K > 0$, $0 < \eta < 1$ by Lemma 2.1. Thus

$$\begin{aligned} |\overline{V}_\alpha(i)| &= \Big| E\Big[\sum_{m=0}^{\infty} \alpha^m c(X_m, v_\alpha(X_m)) \;\Big|\; X_0 = i \Big] \\ &\qquad - E\Big[\sum_{m=0}^{\infty} \alpha^m c(X_m, v_\alpha(X_m)) \;\Big|\; X_0 = 0 \Big] \Big| \\ &= \Big| E\Big[\sum_{m=0}^{\infty} \alpha^m (c(X_m, v_\alpha(X_m)) - \beta_\alpha) \;\Big|\; X_0 = i \Big] \\ &\qquad - E\Big[\sum_{m=0}^{\infty} \alpha^m (c(X_m, v_\alpha(X_m)) - \beta_\alpha) \;\Big|\; X_0 = 0 \Big] \Big| \\ &\le \sum_{m=0}^{\infty} \alpha^m |E[c(X_m, v_\alpha(X_m)) - \beta_\alpha) \mid X_0 = i]| \\ &\qquad + \sum_{m=0}^{\infty} \alpha^m |E[c(X_m, v_\alpha(X_m)) - \beta_\alpha) \mid X_0 = 0]| \\ &\le \frac{2K}{1-\eta} < \infty. \end{aligned} \tag{21}$$

Similarly

$$|(1-\alpha)V_\alpha(0)| = (1-\alpha)\Big| E\Big[\sum_{m=0}^{\infty} \alpha^m c(X_m, v_\alpha(X_m)) \;\Big|\; X_0 = 0 \Big] \Big|$$

$$\le (1-\alpha) \sum_{m=0}^{\infty} \alpha^m |E[(c(X_m, v_\alpha(X_m)) - \beta_\alpha) \mid X_0 = 0]| + \beta_\alpha \le \frac{(1-\alpha)K}{1-\eta} + \beta_\alpha.$$

Thus

$$\limsup_{\alpha\uparrow 1} (1-\alpha)V_\alpha(0) \le \beta_\alpha \le \beta < \infty. \tag{22}$$

In view of (21)–(22), we can invoke the Bolzano-Weierstrass theorem to let $\alpha\uparrow 1$ in (19) and conclude that any limit point $(V'(\cdot), \beta')$ of $(\overline{V}_\alpha(\cdot), (1-\alpha)V_\alpha(0))$ as $\alpha \uparrow 1$ satisfies (13). Clearly, $\overline{V}_\alpha(0) = 0$, hence $V'(0) = 0$. Furthermore, it follows from (8) and the arguments leading to (21) that V' will be $O(x)$. By the uniqueness part of Theorem 4.1, $V' \equiv V$. This completes the proof. ∎

In the next section we leverage these results to establish the Whittle indexability of our problem.

5. Whittle indexability

5.1. Proof of indexability. We shall prove Whittle indexability by establishing some structural properties of the value function V. For this purpose, we view the individual state evolution of a cluster in the following manner:

$$X_{n+1} = X_n - D_n + \nu_n \xi_{n+1}, \qquad n \ge 0, \tag{23}$$

where the arrivals $\{\xi_n\}$ are i.i.d. Poisson and the control $\{\nu_n\}$ non-anticipative as before, but the departure process $\{D_n\}$ is reinterpreted as follows: We assign to each server j a family of i.i.d. $\{0,1\}$-valued random variables $\{\phi_n^j\}$ with the interpretation that $\phi_n^j = 1$ with probability q, implying a potential service completion. By this we mean that in case there is a client being served, her service is completed. If not, it is a dummy event. This framework allows us to fix on a probability space processes $\{\xi_n\}, \{\phi_n^j\}, \{\nu_n\}$ and consider the evolution (23) simultaneously for more than one initial condition. Note that for non-anticipativity of $\{\nu_n\}$, all we need is that $\xi_m, \phi_m, m > n$, be independent of ν_n for all n, where we have dropped the superscript j for ϕ_m for notational ease.

LEMMA 5.1. *V is increasing in its argument.*

Proof. Consider two processes $\{X_n\}, \{X'_n\}$ governed by (23) with common arrival process $\{\xi_n\}$, service completions $\{\phi_n\}$ and non-anticipative control process $\{\nu_n\}$, with initial conditions $y > z$ resp. Then $X_n \ge X'_n$ for all n a.s. Since our reward is increasing in the state variable, it follows that for the α-discounted problem discussed above, we have

$$J_\alpha(z, \{\nu_n\}) \le J_\alpha(y, \{\nu_n\}).$$

Taking the maximum over all non-anticipative $\{\nu_n\}$ on both sides, we get

$$V_\alpha(z) \le V_\alpha(y), \quad \text{so} \quad \bar{V}_\alpha(z) \le \bar{V}_\alpha(y).$$

Letting $\alpha \uparrow 1$ yields the claim in view of Lemma 4.3 above. ∎

LEMMA 5.2. *V has increasing differences in the sense that for $z > 0$ and $x > y$,*

$$V(x+z) - V(x) \geq V(y+z) - V(y).$$

Proof. It suffices to consider $z = 1$ and $y = x - 1$ for $x \geq 1$. As above, construct on a common probability space three processes $\{X'_n\}, \{X_n\}, \{X''_n\}$ with common arrivals, service completions and non-anticipative control process $\{\nu_n\}$, and initial conditions $x+1, x, x-1$ resp. for some $x \geq 1$. The processes $\{X'_n\}, \{X_n\}$ become identical, or couple, once they meet, which occurs the first time a departure occurs which is real for the former and dummy for the latter. The same holds true for the pair $\{X_n\}, \{X''_n\}$. It is clear that the latter coupling will happen before the former, that is, if τ_1, τ_2 are the respective coupling times, then $\tau_1 \geq \tau_2$ a.s. Then

$$\begin{aligned} J_\alpha(x+1, \{\nu_n\}) - J_\alpha(x, \{\nu_n\}) &= E[(p-\mu)(\tau_1 - 1)] \\ &\geq E[(p-\mu)(\tau_2 - 1)] \\ &= J_\alpha(x, \{\nu_n\}) - J_\alpha(x-1, \{\nu_n\}). \end{aligned}$$

Thus

$$2J_\alpha(x, \{\nu_n\}) \leq J_\alpha(x+1, \{\nu_n\}) + J_\alpha(x-1, \{\nu_n\}).$$

Taking the maximum over $\{\nu_n\}$ on both sides, we have

$$2V_\alpha(x) \leq V_\alpha(x+1) + V_\alpha(x-1).$$

Hence

$$2\bar{V}_\alpha(x) \leq \bar{V}_\alpha(x+1) + \bar{V}_\alpha(x-1).$$

Letting $\alpha \uparrow 1$ and using Lemma 4.3 gives

$$2V(x) \leq V(x+1) + V(x-1).$$

This proves the result. ∎

THEOREM 5.1. *The optimal policy is a threshold policy.*

Proof. For the active and passive states to be equally preferred at state x, we need

$$\lambda + k = f(x) := E[V(x - D + \xi)] - E[V(x - D)],$$

where D is a sum of x independent Bernoulli random variables with mean q, and ξ is independent Poisson with mean Λ. Then

$$\begin{aligned} f(x) &= E_x[V(x - D + \xi)] - E_x[V(x - D)] \\ &= \sum_{d=0}^{x} \binom{x}{d} q^d (1-q)^{x-d} \sum_{a=0}^{\infty} \frac{\Lambda^a}{a!} e^{-\Lambda} V(x + a - d) \\ &\quad - \sum_{d=0}^{x} \binom{x}{d} q^d (1-q)^{x-d} V(x - d) \end{aligned} \tag{24}$$

$$= \sum_{d=0}^{x} \binom{x}{d} q^d (1-q)^{x-d} \left(\sum_{a=0}^{\infty} \frac{\Lambda^a}{a!} e^{-\Lambda} V(x+a-d) - V(x-d) \right)$$
$$= \sum_{d=0}^{x} \binom{x}{d} q^d (1-q)^{x-d} \sum_{a=0}^{\infty} \frac{\Lambda^a}{a!} e^{-\Lambda} (V(x+a-d) - V(x-d))$$
$$= \sum_{d=0}^{x} \binom{x}{d} q^d (1-q)^{x-d} B(x-d)$$

where

$$B(x) = \sum_{a=0}^{\infty} \frac{\Lambda^a}{a!} e^{-\Lambda} (V(x+a) - V(x)).$$

By Lemma 5.2, $B(x+1) \geq B(x)$. For the optimal policy to be threshold we need $f(x+d) \geq f(x)$ for $d > 0$. We have

$$f(x+1) = \sum_{d=0}^{x+1} \binom{x+1}{d} q^d (1-q)^{x+1-d} B(x+1-d) \tag{25}$$
$$= (1-q)^{x+1} B(x+1) + \sum_{d=1}^{x+1} \binom{x+1}{d} q^d (1-q)^{x+1-d} B(x+1-d)$$
$$= (1-q)^{x+1} B(x+1) + \sum_{d=0}^{x} \binom{x+1}{d+1} q^{d+1} (1-q)^{x-d} B(x-d).$$

Subtracting (24) from (25), we get

$$f(x+1) - f(x)$$
$$= (1-q)^{x+1} B(x+1)$$
$$+ \sum_{d=0}^{x} \left(\binom{x+1}{d+1} q^{d+1} (1-q)^{x-d} B(x-d) - \binom{x}{d} q^d (1-q)^{x-d} B(x-d) \right)$$
$$= (1-q)^{x+1} B(x+1) + \sum_{d=0}^{x} \binom{x}{d} q^d (1-q)^{x-d} \left(\frac{(x+1)q}{d+1} - 1 \right) B(x-d)$$
$$= (1-q)^{x+1} B(x+1) - q^x (1-q) B(0) \tag{26}$$
$$+ \sum_{d=0}^{x-1} \binom{x}{d} q^d (1-q)^{x-d} \left(\frac{(x+1)q}{d+1} - 1 \right) B(x-d)$$
$$= (1-q)^{x+1} B(x+1) - q^x (1-q) B(0) \tag{27}$$
$$+ \sum_{d=0}^{x-1} \left(\binom{x}{d+1} q^{d+1} (1-q)^{x-d} - \binom{x}{d} q^d (1-q)^{x+1-d} \right) B(x-d)$$

$$
\begin{aligned}
&= \left\{(1-q)^{x+1}B(x+1) + \sum_{d=0}^{x-1}\binom{x}{d+1}q^{d+1}(1-q)^{x-d}B(x-d)\right\} \\
&\quad - \left\{q^x(1-q)B(0) + \sum_{d=0}^{x-1}\binom{x}{d}q^d(1-q)^{x+1-d}B(x-d)\right\} \\
&= \left\{(1-q)^{x+1}B(x+1) + \sum_{d=1}^{x}\binom{x}{d}q^d(1-q)^{x+1-d}B(x+1-d)\right\} \\
&\quad - \sum_{d=0}^{x}\binom{x}{d}q^d(1-q)^{x+1-d}B(x-d) \\
&= \sum_{d=0}^{x}\binom{x}{d}q^d(1-q)^{x+1-d}B(x+1-d) - \sum_{d=0}^{x}\binom{x}{d}q^d(1-q)^{x+1-d}B(x-d) \\
&= (1-q)\sum_{d=0}^{x}\binom{x}{d}q^d(1-q)^{x-d}\big(B(x+1-d) - B(x-d)\big) \\
&\geq 0
\end{aligned}
$$

as desired. The passage from (26) to (27) is given in the Appendix. Thus f is increasing. Since the set of passive states is

$$\{x : \lambda \geq -k + f(x)\},$$

it follows that the optimal policy is a threshold policy. This policy is such that below the threshold the queue remains passive, and above the threshold it remains active. ∎

Observe that below the threshold, there are no arrivals, only departures. Thus the chain gets absorbed into the zero state. Since the chain is stable, even from an active state it will eventually hit the passive state and subsequently get absorbed into the zero state. Thus it is transient with a single absorbing state, an extreme case of the uni-chain condition wherein the single communicating class is in fact a singleton. The zero state being passive, the constant reward of λ is received once absorbed there, thus $\beta = \lambda$.

THEOREM 5.2. *The server allocation problem is Whittle indexable.*

Proof. We now need to prove that the threshold is increasing. We already have increasing differences in x, i.e., for $a_1, a_2 > 0$,

$$
\begin{aligned}
&V(\lambda, x + a_1 + a_2) - V(\lambda, x + a_1) \geq V(\lambda, x + a_2) - V(\lambda, x) \\
&\qquad \Rightarrow\ E[V(x - D + \xi)] - E[V(x - D)] \text{ increases with } x \\
&\qquad \Rightarrow\ \text{threshold policy: } \exists\ x^*(\lambda) \text{ such that } x < x^*(\lambda) \ \Leftrightarrow\ \text{passive.}
\end{aligned}
$$

Let τ_0 be the first hitting time of 0 as before. Consider a threshold policy.

For $x < \textit{threshold}$,

$$V(\lambda, x) = \lambda E_x[\tau_0] + E_x\Big[\sum_{m=0}^{\tau_0}(p-\mu)X_m\Big] - \beta(\lambda)E_x[\tau_0].$$

For $x > \textit{threshold}$,

$$V(\lambda, x) = E_x\Big[\sum_{m=0}^{\tau_0}(p-\mu)X_m\Big] + \lambda E_x[\tau_0 - \sigma] - kE_x[\sigma] - \beta(\lambda)E_x[\tau_0],$$

where σ is the first time the state drops below the threshold.

We know that $\beta(\lambda) = \lambda$. Substituting this we get

$$V(\lambda, x) = E_x\Big[\sum_{m=0}^{\tau_0}(p-\mu)X_m\Big]$$

in the former case, and

$$\begin{aligned} V(\lambda, x) &= E_x\Big[\sum_{m=0}^{\tau_0}(p-\mu)X_m\Big] - (k+\lambda)E_x[\sigma] \\ &= E_x\Big[\sum_{m=0}^{\sigma-1}(p-\mu)X_m\Big] - (k+\lambda)E_x[\sigma] + E_x[V(\lambda, X_\sigma)] \end{aligned}$$

in the latter. The quantity inside the expectation in the third term above is the value function once the state goes below the threshold, and is equivalent to the value function for the passive state $X(\sigma)$ since there will not be any more arrivals. Let $\lambda_1 < \lambda_2$ with the corresponding thresholds x_1, x_2 resp. Suppose $x_1 > x_2$. Consider a single sample path. Fix the realizations of ϕ_m, ξ_m, $m \geq 0$, and consider the dynamics with the two initial conditions x_1, x_2, leading to processes $\{X_m^1\}, \{X_m^2\}$ resp. with $X_m^1 \geq X_m^2$ for all m a.s. In either case, σ is either 1 if $\xi_0 \leq D_0$, or equals the first time the random walk $Z_n := \sum_{m=1}^{n}(\xi_m - D_m)$, $m \geq 1$, drops by at least $\xi_0 - D_0$ otherwise. In any case, it is identical for both the processes. A term by term comparison of

$$\begin{aligned} \lambda_1 = E\Big[\sum_{m=0}^{\sigma-1}(p-\mu)X_m^1\Big] &- E[\sigma](k+\lambda_1) + E[V(\lambda_1, X_\sigma^1)] \\ &- E[V(\lambda_1, x_1 - D_0)] - k \end{aligned}$$

and

$$\begin{aligned} \lambda_2 = E\Big[\sum_{m=0}^{\sigma-1}(p-\mu)X_m^2\Big] &- E[\sigma](k+\lambda_2) + E[V(\lambda_2, X_\sigma^2)] \\ &- E[V(\lambda_2, x_2 - D_0)] - k \end{aligned}$$

yields:

- $E\sum_{m=0}^{\sigma-1}(p-\mu)X_m^1] > E[\sum_{m=0}^{\sigma-1}(p-\mu)X_m^2]$, because $X_m^1 \geq X_m^2$ for all m a.s. with strict inequality for $m=0$.
- $-E[\sigma](k+\lambda_1) > -E[\sigma](k+\lambda_2)$, because $\lambda_2 > \lambda_1$.
- $E[V(\lambda_1, X_1(\sigma))] - E[V(\lambda_1, x_1 - D_0)] \geq E[V(\lambda_2, X_2(\sigma))] - E[V(\lambda_2, x_2 - D_0)]$, by virtue of the following reasoning: Observe that both the LHS and RHS are a difference of two terms. Both these terms are the value function for passive states with no arrivals and fixed identical departures. In this range,

$$V(\lambda_i, x) = E\Big[\sum_{m=0}^{\tau_0}(\lambda_i + (p-\mu)X_m^i - \beta_i) \;\Big|\; X_0^i = x\Big]$$
$$= E\Big[\sum_{m=0}^{\tau_0}(p-\mu)X_m^i \;\Big|\; X_0^i = x\Big]$$

because $\beta_i = \lambda_i$. Thus it is independent of λ_i. Next, note that X_σ^i is arrived at by a single departure from an active state, whereas $x_i - D_0$ is arrived at by a single departure from the threshold. Hence the former stochastically dominates the latter. Using a well known consequence of stochastic dominance [17], we can realize replicas in law of the two random variables on a common probability space so that the former is no smaller than the latter a.s. Given the irrelevance of λ_i noted above, the claim then follows from Lemma 5.2.

Combining the above, we get $\lambda_1 > \lambda_2$. This is a contradiction. Thus the threshold λ increases with x. As a result, the passive set increases from ϕ to S as λ varies from $-\infty$ to ∞, and the problem is Whittle indexable. ∎

5.2. Computation of Whittle indices. Fix x. The Whittle index $\lambda(x)$ for x is the value of λ for which active and passive modes are equally preferred at x, i.e., x is the threshold between activity and passivity. Then the state evolution equation becomes

$$X_{n+1} = X_n - D_n + I\{X_n > x\}\xi_{n+1}, \qquad n \geq 0.$$

Here D_n is the sum of X_n i.i.d. Bernoulli random variables with mean q, and ξ_n are i.i.d. Poisson with mean Λ. Given that this is the optimal process, we may consider the associated Poisson equation in lieu of the dynamic programming equation, i.e.,

$$V(y) = (p-\mu)y - k - \beta + \sum_z p_a(z\,|\,y)V(z), \qquad y > x, \tag{28}$$

$$V(y) = (p-\mu)y + \lambda - \beta + \sum_z p_b(z\,|\,y)V(z), \qquad y \leq x, \tag{29}$$

$$V(0) = 0. \tag{30}$$

This has a unique solution $(V(\cdot), \beta)$ where V is $O(y)$. However, we also want λ to be the Whittle index for state x, i.e., the active and passive states should be equally desirable at x for this value of λ. Then from the dynamic programming equation,

$$\lambda = -k + \sum_y p_a(y \mid x)V(y) - \sum_y p_b(y \mid x)V(y), \tag{31}$$

which is not automatic. This suggests an iterative scheme that updates an estimate λ_n for the λ satisfying the above by incrementally decreasing, resp. increasing it when the RHS is too low, resp. high. The scheme we use is the simple adaptation given by

$$\lambda_{n+1} = \lambda_n + \gamma\Big(-k + \sum_y p_a(y \mid x)V_{\lambda_n}(y) - \sum_y p_b(y \mid x)V_{\lambda_n}(y) - \lambda_n\Big), \quad n \geq 0, \tag{32}$$

where $\gamma > 0$ is a small stepsize and $(V_\lambda, \beta(\lambda))$ is the unique solution to (28)–(29) for a given λ. This iteration starts with a guess of λ and incrementally corrects it in favour of satisfying (31). Since γ is small, we can view the iteration as an Euler scheme for the o.d.e.

$$\dot{\lambda}(t) = -k + \Big(\sum_y p_a(y \mid x)V(\lambda(t), y) - \sum_y p_b(y \mid x)V(\lambda(t), y)\Big) - \lambda(t), \quad t \geq 0. \tag{33}$$

A calculation analogous to that in the proof of Theorem 5.2 shows that the quantity in large brackets is a function of the form $A - B\lambda$, thus (33) is a simple stable linear differential equation which converges to its unique equilibrium. This ensures the approximate convergence of its Euler approximation (32). Since we are interested in ordinal comparison of indices, a small error (smallness ensured by using a small γ) is tolerable.

6. Numerical experiments

6.1. Calculation of Whittle index. We solve equation (28)–(31) to obtain Whittle index of each price cluster (as a function of its state). We get $\beta = \lambda$ by substituting (30) in (29). Various parameters in our experiments are set as follows:

- $p_1 = 22$, $p_2 = 35$, $p_3 = 49$,
- $q = 0.3$, $\mu = 5$, $\gamma = 0.01$,
- $\Lambda_1 = 0.23$, $\Lambda_2 = 0.16$, $\Lambda_3 = 0.12$,
- $k_i = 50(\Lambda_1 - \Lambda_i)$ for $i \in \{1, 2, 3\}$.

Figure 4 shows the Whittle index $\lambda_i(x)$ for various values of x and different pricing options. To decide what price to operate at $n+1$ for a given state X_n^i for all i, we compare the different $\lambda_i(X_n^i)$ and pick the one that is maximum.

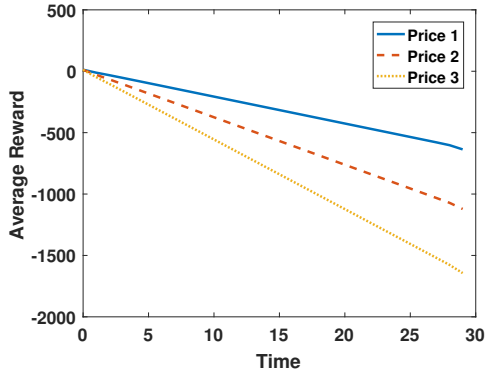

Fig. 4. Whittle indices

Due to the decreasing nature of the indices, if a particular cluster is operated for a long time, its Whittle index would reduce, leading to some other cluster being operated.

6.2. Comparison with Greedy scheme. We consider three price levels and simulate Poisson arrivals by allowing arrival rates to decrease with price. Departures are Bernoulli with parameter q as above. We compare the overall revenue for two pricing schemes—the greedy and the Whittle. At any instant t, the greedy scheme selects the cluster with the highest average reward up to t, whereas the Whittle scheme chooses the price corresponding to the cluster with the maximum value of Whittle index $\lambda(x)$ where x is the state of the cluster at time t. We simulated the system for 10000 time units using random price exploration for the first 3000 time units and invoking the above price schemes thereafter.

Figure 5 shows the time average reward computed at individual prices for the greedy scheme. It can be seen that the queue which had the maximum reward at the end of the exploration phase continues to be chosen subsequently since its average reward dominates that of others; in other words, the system gets locked in that price. The figure also shows the time average reward for all the prices in the case of Whittle scheme. As can be seen, different prices continue to be active at different time points avoiding lock-in to any particular price. Particularly, the cluster that has *vanished* or about to vanish will again become active under the Whittle scheme. The same conclusion can be drawn from the cumulative reward plots in Figure 5. Notice that in the cases of p_1 and p_3, cumulative reward for the greedy scheme does not change after a certain time, as these price clusters remain empty with no further arrivals. In contrast, in the Whittle scheme, cumulative rewards for all the prices continue to increase (albeit at different rates), indicating that they are active at various points.

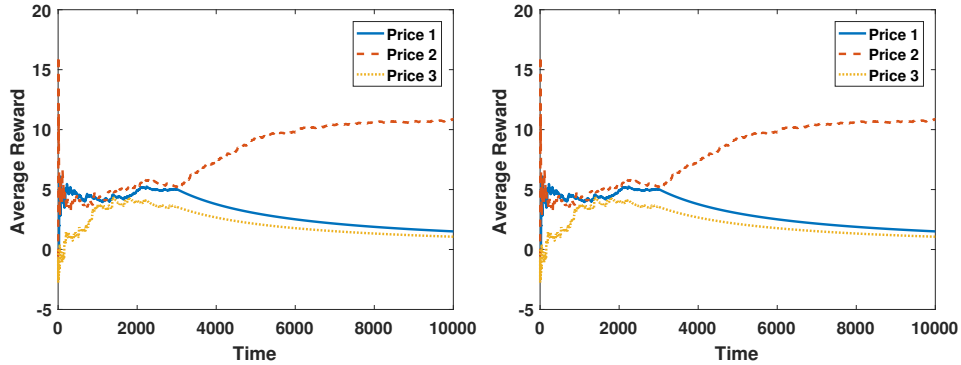

Fig. 5. Average rewards of Greedy and Whittle

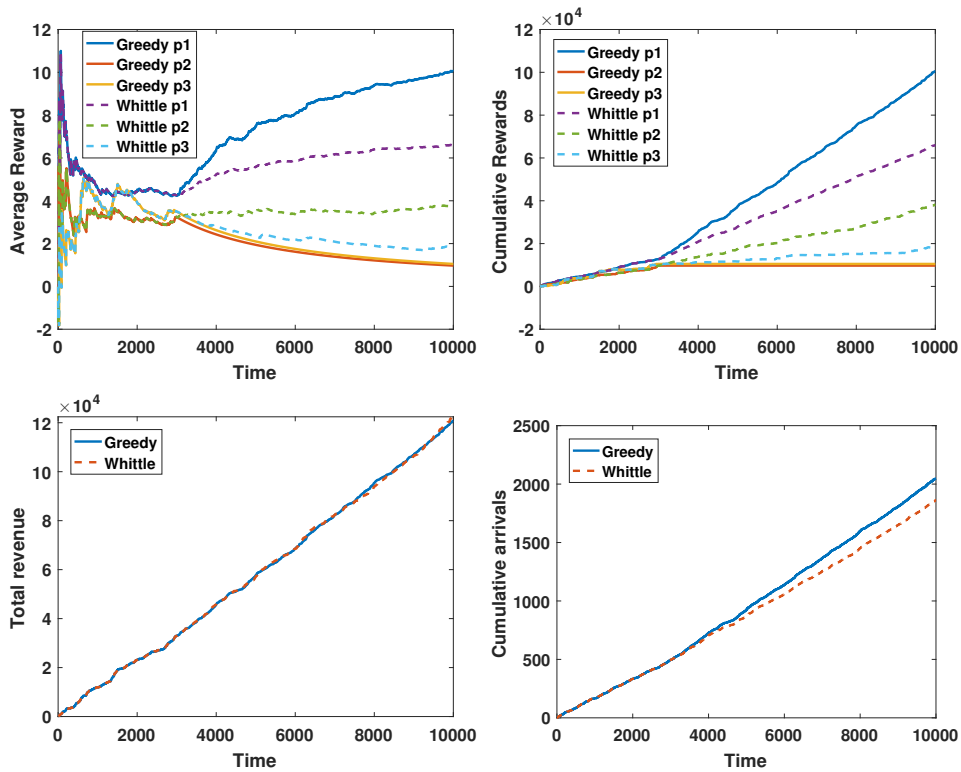

Fig. 6. Average reward, cumulative reward, cumulative revenue, and cumulative arrivals for the greedy scheme locked in Price 1

Depending on which price the greedy scheme gets locked in, revenue performance of the Whittle scheme can be better or worse than that of the greedy policy. Figures 6–8 present the cases when the greedy policy gets locked in low, medium and high prices (p_1, p_2, p_3) respectively. Dominance

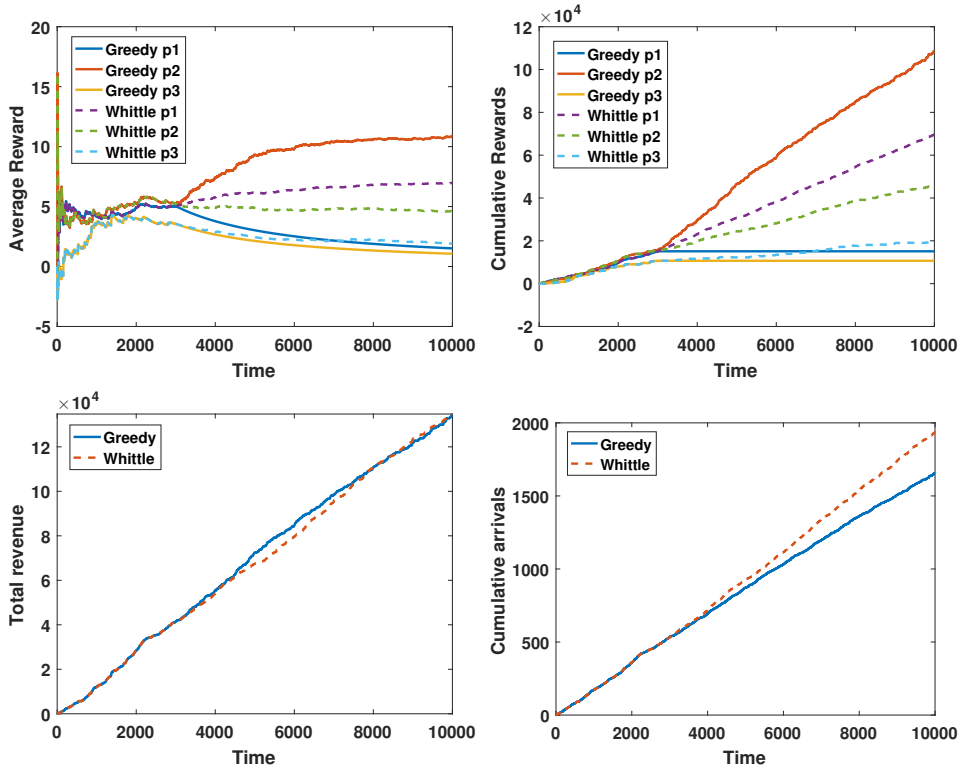

Fig. 7. Average reward, cumulative reward, cumulative revenue, and cumulative arrivals for the greedy scheme locked in Price 2

of one scheme over the other with respect to total revenue is dependent on the price sensitivity of the arrival process. Note that since the departure rate is assumed not to depend on price, price commitment does not impart any stickiness or extended usage even at lower prices. In the absence of such service stickiness, our experimented inverse demand curve can be observed to demonstrate convexity, and thus higher sensitivity at lower price values. When the greedy scheme gets locked in the highest price p_3, the arrival rate is the lowest, and hence the average number of jobs that last longer is also minimal, reducing the potential for revenue generation in future. In contrast, the Whittle scheme activates lower prices to take advantage of more dense arrivals at those prices, and thus extract potential revenues from a higher average number of jobs with longer service durations. Similarly, when the greedy policy is locked in the lowest price, the Whittle scheme explores the opportunity for higher revenues charging higher prices per unit time. Figures 6–8 clearly support this intuition. It is interesting to note that the Whittle scheme often serves a greater number of customers, offering increased uti-

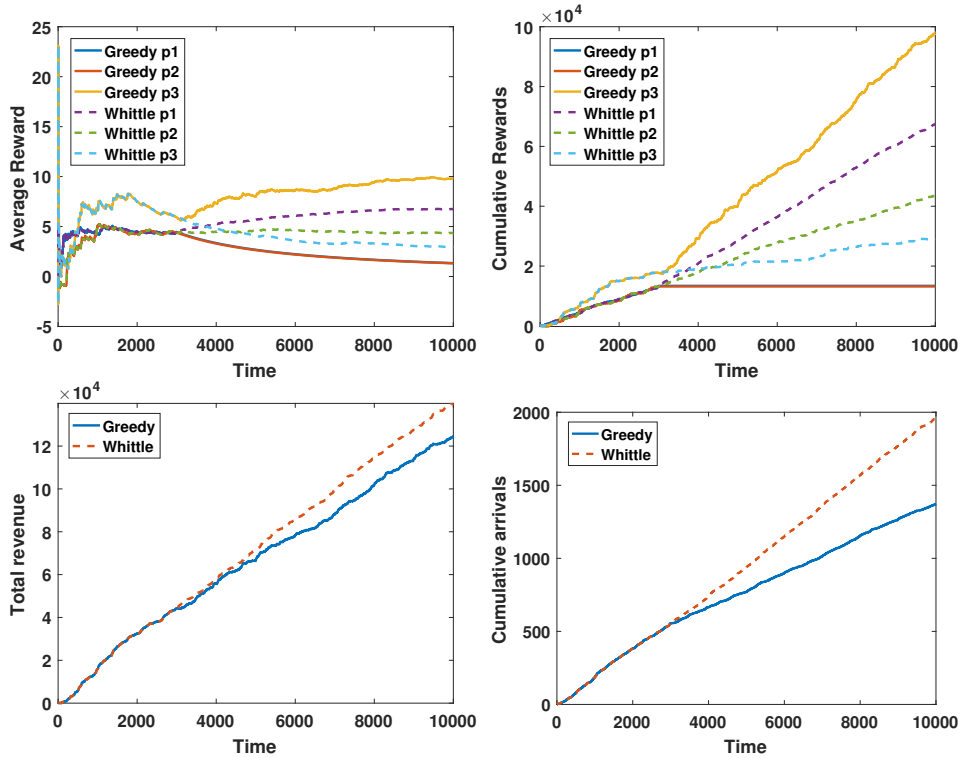

Fig. 8. Average reward, cumulative reward, cumulative revenue, and cumulative arrivals for the greedy scheme locked in Price 3

lization of the physical server—the higher the number of jobs served, the higher the utilization because usage levels of all the jobs are i.i.d. and price independent. The cumulative arrival plots of Figures 6–8 demonstrate high utilization levels of the Whittle scheme. In short, the Whittle scheme strikes a balance between revenue and server utilization.

Table 1 shows the results for 100 simulations of the above schemes. 'Frequency' is the number of times the greedy policy got locked in a particular price; 'Difference in arrivals' is the difference in the total number of customers served by the Whittle policy compared to the greedy policy; 'Total

Table 1. Comparison of Whittle and greedy policies for different lock-in prices of the greedy policy

	Frequency	Difference in arrivals	Total revenue difference (%)
p_1	41	−218.8	−0.318
p_2	27	269	0.432
p_3	32	559.8	3.527

revenue difference' is the percentage difference between the total revenue of the Whittle policy compared to Greedy policy. It can be seen that the revenue difference is within $\pm 1\%$ for the first two cases, while it is $> 3\%$ for the last case. The distribution of the total revenue difference for different lock-in prices can also be seen in Figure 9. It can be seen that when the greedy policy gets locked in p_3, the Whittle policy gains more revenue 80% of the time, and further 60% of the time the percentage difference is more than 2% (up to 11–13% usually). As a result, the overall revenue difference is 1.11% and the average difference in cumulative arrivals is 162, demonstrating that the Whittle policy improves server utilization as well.

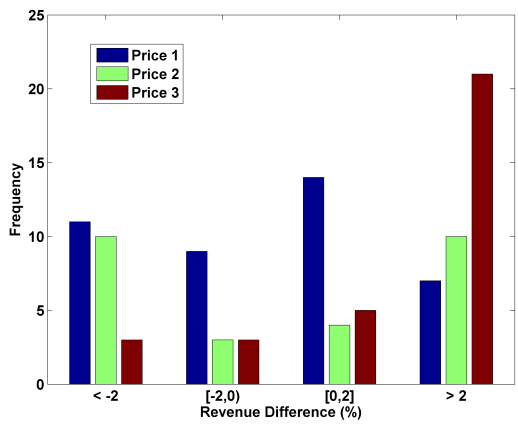

Fig. 9. The distribution of the total revenue difference (%) at different lock-in prices of the greedy policy

6.3. Comparison with a periodic pricing scheme. We compare the Whittle policy with a periodic policy wherein each price is repeated after a specific duration ($\approx$ 330 time units in our experiments). Whittle policy again outperforms the periodic policy on the average. The typical average rewards per price, cumulative rewards per price, total cumulative revenue and the cumulative arrivals for both the policies are shown in Figure 10. The periodic nature of the periodic policy is evident from the step like nature of its cumulative revenue per price curves.

The percentage total revenue difference between the Whittle and the periodic policy was found for 100 simulations and the results are shown in Figure 11. Again, it can be seen that 70% of the time, the revenue is more for the Whittle policy with 1.83% gain over the periodic policy on the average. Cumulative arrivals under the Whittle policy are, on the average, 193.5 units higher than those under the periodic policy, with the Whittle policy dominating always. Thus, the Whittle policy not only serves more customers but also generates higher total revenue.

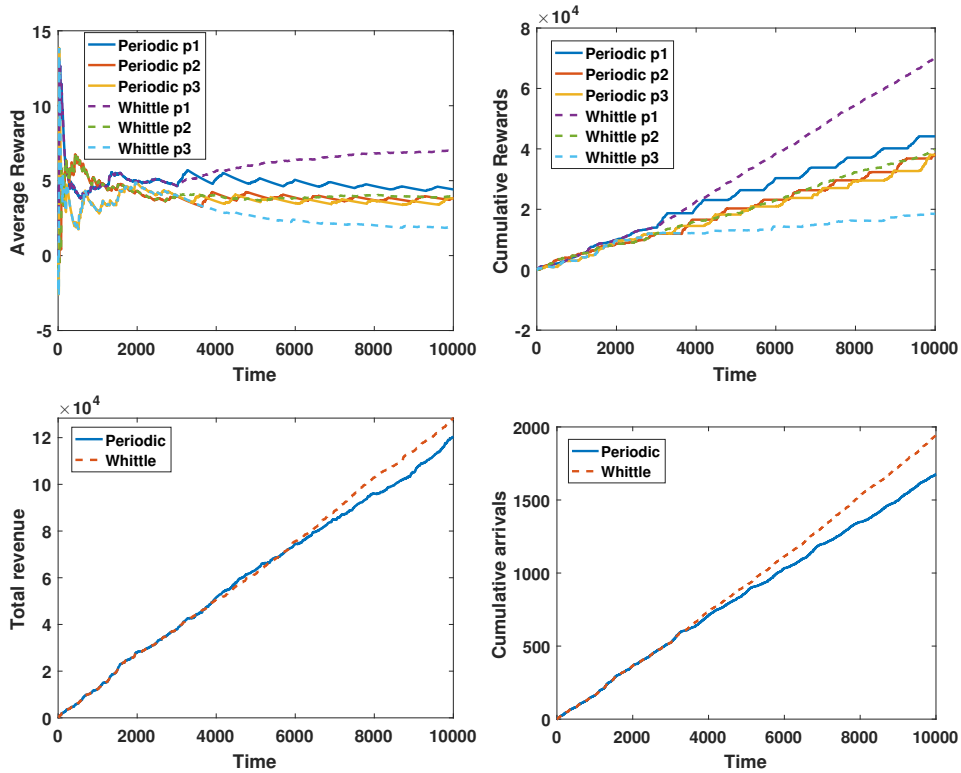

Fig. 10. Average reward, cumulative reward, cumulative revenue, and cumulative arrivals for the periodic policy

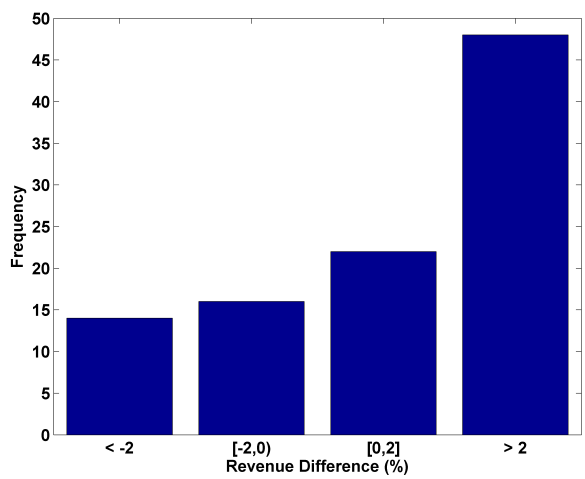

Fig. 11. The distribution of the percentage revenue difference between the Whittle scheme and the periodic policy

By comparing the differences in performance between the Whittle policy and other policies, one can conclude that Whittle policy > greedy policy > periodic policy when the average total revenue is considered as a metric.

7. Conclusion. We have proposed a dynamic pricing scheme based on Whittle's index for pricing the cloud infrastructure service under price commitment offers. In such an environment, jobs arrive with a pre-decided price expectation and join for service if the price being offered is lower than or equal to their price expectation. A job is served without preemption and is charged the same price rate at which it joined until its service completion. We proved Whittle indexability of the problem and proposed the use of Whittle index to decide which price to operate at. Our simulation study clearly demonstrated superiority of the Whittle index policy over two intuitive policies, namely the greedy policy and a periodic pricing policy, with regard to physical server utilization and revenue generation. The parameters of the problem being coupled, a further study is needed to find the best possible (arrival rate, price) combination to meet a desired revenue expectation.

From the modeling perspective, an important extension to the problem is to analyze dynamic pricing under *jockeying*, wherein price sensitive or opportunistic customers may abruptly terminate their ongoing service and rejoin the system when the price offered is low. Customers may adopt such strategies on non-critical jobs.

We highlight here some possible technical extensions to our work. We took μ as a given penalty. If we treat it as a formal Lagrange multiplier, we can iteratively compute it by a gradient ascent scheme

$$\mu_{n+1} = \mu_n + \kappa\Big(\sum_i X_n^i - M\Big), \qquad n \geq 0,$$

where $\kappa > 0$ is a small step size and $\{X_n^i\}$ are simulated according to the Whittle index policy with $\mu = \mu_n$. This can be justified using a 'two time scale' argument. We do not get into the details here. It still remains a heuristic because the Whittle index policy itself is, and appears amenable only as an off-line scheme. More research in this direction is needed for an informed choice of μ.

The fact that the decoupled control problems end up as transient chains is unusual in applications of Whittle index and questions may be raised about validity of using the average reward dynamic programming equation. One way to justify this is to consider an irreducible perturbation, e.g., at each time using the index policy with probability $1 - p$ and using active or passive with equal probabilities $= p/2$, where p is very small. One can then write a legitimate average reward dynamic programming equation and then

let $p \downarrow 0$ to recover the dynamic programming equation employed here. This is in the spirit of 'viscosity solutions' in deterministic control.

We conclude with some comments regarding discounted reward problems. While distinct in flavor from the average cost framework we have used (as has Whittle in [20]), it too has been a popular criterion for restless bandits. While there is no a priori issue with introducing the Whittle subsidy λ paving way towards a definition of Whittle indexability and Whittle index, to have a clean motivation by analogy with the Lagrange multiplier as in [20], one would have to consider the discounted reward with the corresponding constraint on the expected discounted sum of $\{\nu_n\}$. There has also been some interesting recent work generalizing the classical discounted case by (among other generalizations) allowing for different discount factors. There is no exact counterpart for the average cost we consider, but a mix of average or discounted costs with different discount factors depending on the bandit is an immediate possibility suggested by [20] that might be worth pursuing.

8. Appendix. We now show that (26) implies (27). From (26), we have

$$\begin{aligned}
&\binom{x}{d} q^d (1-q)^{x-d} \left(\frac{(x+1)q}{d+1} - 1 \right) \\
&\qquad = \binom{x}{d} q^d (1-q)^{x-d} \left(\frac{(x+1)q \pm qd}{d+1} - 1 \right) \\
&\qquad = \binom{x}{d} q^d (1-q)^{x-d} \left(\frac{(x-d)q + (d+1)q}{d+1} - 1 \right) \\
&\qquad = \binom{x}{d} q^d (1-q)^{x-d} \left(\frac{(x-d)q}{d+1} - 1 + q \right) \\
&\qquad = \binom{x}{d} q^d (1-q)^{x-d} \left(\frac{(x-d)q}{d+1} - (1-q) \right) \\
&\qquad = \binom{x}{d} q^d (1-q)^{x-d} \frac{(x-d)q}{d+1} - \binom{x}{d} q^d (1-q)^{x-d} (1-q) \\
&\qquad = \frac{x!}{(x-d)!d!} \frac{(x-d)}{d+1} q^{d+1} (1-q)^{x-d} - \binom{x}{d} q^d (1-q)^{x+1-d} \\
&\qquad = \frac{x!}{(x-d-1)!(d+1)!} q^{d+1} (1-q)^{x-d} - \binom{x}{d} q^d (1-q)^{x+1-d} \\
&\qquad = \binom{x}{d+1} q^{d+1} (1-q)^{x-d} - \binom{x}{d} q^d (1-q)^{x+1-d},
\end{aligned}$$

as desired.

Acknowledgments. Research of V. S. Borkar was supported in part by CEFIPRA grant 5100-IT-1.

References

[1] E. Altman, *Constrained Markov Decision Processes*, CRC Press, Boca Raton, FL, 1998.

[2] O. A. Ben-Yehuda, M. Ben-Yehuda, A. Schuster, and D. Tsafrir, *Deconstructing Amazon EC2 spot instance pricing*, ACM Trans. Economics Comput. 1 (2013), no. 3, art. 16.

[3] V. S. Borkar, *Topics in Controlled Markov Chains*, Pitman Res. Notes in Math. 240, Longman Sci. Tech., Harlow, 1991.

[4] V. S. Borkar, *Convex analytic methods in Markov decision processes*, in: Handbook of Markov Decision Processes: Methods and Applications, E. A. Feinberg and A. Shwartz (eds.), Kluwer, Boston, MA, 2002, 347–375.

[5] V. S. Borkar, *Uniform stability of controlled Markov processes*, in: System Theory: Modeling, Analysis and Control, T. E. Djaferis and I. C. Schick (eds.), Kluwer, Boston, MA, 2009, 107–120.

[6] W. Cowan and M. N. Katehakis, *Multi-armed bandits under general depreciation and commitment*, Problems in Engrg. Information Sci. 29 (2015), 51–76.

[7] J. Gittins, K. Glazebrook, and R. Weber, *Multi-Armed Bandit Allocation Indices*, 2nd ed., Wiley, Chichester, 2011.

[8] P. Jacko, *Dynamic Priority Allocation in Restless Bandit Models*, Lambert, 2010.

[9] K. Liu and Q. Zhao, *Indexability of restless bandit problems and optimality of Whittle index for dynamic multichannel access*, IEEE Trans. Information Theory 56 (2010), 5547–5567.

[10] S. P. Meyn and R. L. Tweedie, *Markov Chains and Stochastic Stability*, 2nd ed., Cambridge Univ. Press, Cambridge, 2009.

[11] J. Niño-Mora and S. S. Villar, *Sensor scheduling for hunting elusive hiding targets via Whittle's restless bandit index policy*, in: Proc. NetGCoop 2011 (Paris, 2011), IEEE, 2011, 8 pp.

[12] J. L. Ny, M. Dahleh, and E. Feron, *Multi-UAV dynamic routing with partial observations using restless bandit allocation indices*, in: Proc. Amer. Control Conf. (ACC 2008) (Seattle, 2008), 4220–4225.

[13] C. H. Papadimitriou and J. N. Tsitsiklis, *The complexity of optimal queuing network control*, Math. Oper. Res. 24 (1999), 293–305.

[14] M. I. Puterman, *Markov Decision Processes*, Wiley, Hoboken, NJ, 1994.

[15] V. Raghunathan, V. S. Borkar, M. Cao, and P. R. Kumar, *Index policies for real-time multicast scheduling for wireless bradcast systems*, Proc. IEEE INFOCOM 2008 (Phoenix, 2008), 2243–2251.

[16] D. Ruiz-Hernandez, *Indexable Restless Bandits*, VDM Verlag, 2008.

[17] M. Shaked and J. G. Shanthikumar, *Stochastic Orders*, Springer, New York, 2007.

[18] R. Sznajder and J. A. Filar, *Some comments on a theorem of Hardy and Littlewood*, J. Optim. Theory Appl. 75 (1992), 201–208.

[19] R. R. Weber and G. Weiss, *On an index policy for restless bandits*, J. Appl. Probab. 27 (1990), 637–648.

[20] P. Whittle, *Restless bandits: activity allocation in a changing world*, J. Appl. Probab. 25 (1988), Special Issue A, 287–298.

V. S. Borkar
Department of Electrical Engineering
Indian Institute of Technology Bombay
Powai, Mumbai 400076, India
E-mail: borkar.vs@gmail.com

K. Ravikumar
TCS Innovation Labs
Cincinnati, OH 45150, U.S.A.
E-mail: ravikumar.karumanchi@tcs.com

Krishnakant Saboo
Department of Electrical Engineering
Indian Institute of Technology Bombay
Powai, Mumbai 400076, India
Current address:
Department of Electrical and Computer Engineering and
the Coordinated Sciences Laboratory
University of Illinois at Urbana-Champaign
Urbana, IL 61801, U.S.A.
E-mail: kishansaboo.2004@gmail.com