Dynamic scheduling in a partially fluid, partially lossy queueing system

Kiran Chaudhary
IEOR, IIT Bombay
kiran30@iitb.ac.in

Veeraruna Kavitha
IEOR, IIT Bombay
vkavitha@iitb.ac.in

Jayakrishnan Nair
EE, IIT Bombay
jayakrishnan.nair@ee.iitb.ac.in

***Abstract*—We consider a single server queueing system with two classes of jobs: *eager* jobs with small sizes that require service to begin almost immediately upon arrival, and *tolerant* jobs with larger sizes that can wait for service. While blocking probability is the relevant performance metric for the eager class, the tolerant class seeks to minimize its mean sojourn time. In this paper, we discuss the performance of each class under dynamic scheduling policies, where the scheduling of both classes depends on the instantaneous state of the system. This analysis is carried out under a certain fluid limit, where the arrival rate and service rate of the eager class are scaled to infinity, holding the offered load constant. Our performance characterizations reveal a *(dynamic) pseudo-conservation law* that ties the performance of both the classes to the standalone blocking probabilities of the eager class. Further, the performance is robust to other specifics of the scheduling policies. We also characterize the Pareto frontier of the achievable region of performance vectors under the same fluid limit, and identify a (two-parameter) class of *Pareto-complete* scheduling policies.**

I. Introduction

In this paper, we analyse a single server queueing system with two heterogeneous customer classes. One class of customers is *eager*—they require service to commence (almost) immediately upon arrival. The performance of the eager class is captured by the blocking probability, i.e., the long run fraction of eager customers that are blocked. The second class of customers is *tolerant*—these customers can tolerate delays and may be queued. The performance of this class is captured via the mean response time of the tolerant customers.

Service systems of this kind are motivated by modern cellular networks, which handle voice calls (which must be either admitted or dropped upon arrival) as well as data traffic (which can be queued). However, such part-loss, part-queueing multi-class service systems are analytically intractable even under the simplest scheduling disciplines (see [1] and the references therein). In this paper, we derive tractable approximations of the performance experienced by each class using a certain fluid limit, referred to as the short-frequent-jobs (SFJ) limit.

The SFJ limit corresponds to scaling the arrival rate as well as the service rate of the eager class to infinity, such that the offered load is held constant. This gives rise to a time-scale separation between the two classes, with the eager class operating at a faster time-scale. Under the SFJ limit, we obtain a closed form characterization of the performance of both classes under a broad class of dynamic scheduling policies that allow the admission control of eager class and scheduling of the eager and tolerant class to be dependent on the size of the tolerant queue.[1] Interestingly, a dynamic pseudo-conservation law follows from this characterization—the performance of both classes depends only on the standalone blocking probabilities (resulting when a single eager scheduling policy is used oblivious to the tolerant state) associated with the eager scheduling schemes employed for each occupancy level of the tolerant queue. In particular, the performance does not depend on the specific scheduling policies that generate those blocking probabilities, as well as on the details of the tolerant scheduler (subject to work conservation and serial, non-anticipative processing). Conservation laws typically allow one to compute the performance of a complex system in terms of the performance of simpler ones. In our case, once the relevant standalone blocking probabilities are known (these can usually be computed easily as they result from the analysis of a single-class loss system), one can compute the performance of both the classes.

We further analyse the Pareto frontier of the performance vectors achievable under the class of dynamic schedulers, which defines the set of efficient operating points for the system. Remarkably, we are able to identify a *Pareto-complete* family of scheduling policies (a family of schedulers is Pareto-complete if it spans the entire Pareto frontier over its parameter space). This family, parametrized by (L, d), where $L \in \mathbb{N}$ and $d \in (0, 1)$, blocks eager customers with the minimum blocking probability when the tolerant occupancy is less than L, with probability d when the occupancy equals L, and with the maximum blocking probability when it exceeds L.

Finally, via numerical experiments, we show that our performance characterizations under the SFJ limit are extremely accurate in the pre-limit (i.e., for moderate values of arrival and service rates of the eager class). This shows that our approximations, which are provably accurate under the SFJ fluid limit, are also useful in practice.

The remainder of this paper is organised as follows. We conclude this introduction with a survey of the related literature. We describe our system model and state some preliminary results in Section II. Under the SFJ limit, we characterize the performance of the tolerant class in Section III, and that of the eager class in Section IV. We formally define the dynamic

The third author acknowledges support from DST and CEFIPRA.

[1] From here on, we follow the convention that admission control (if used) is included in the eager scheduling policy.

achievable region in Section V, and demonstrate the Pareto-complete family of dynamic schedulers in Section VI.

Related literature: The present paper is a follow-up of our prior work [2], [3], which analyses the same heterogeneous queueing system under the SFJ limit for a class of (partially) static scheduling policies. Under this class of policies, the scheduling of the eager class is oblivious to the state of the tolerant queue, with the tolerant queue simply utilizing the service capacity left unused by the eager class. Clearly, this class of schedulers is restrictive. In the present paper, we consider general dynamic policies, where eager scheduling depends on the occupancy of the tolerant queue. This generalization, which requires a non-trivial analysis, results is a substantial expansion of the achievable region of feasible performance vectors (as is shown in Sections V, VI). Moreover, the generalization to dynamic policies necessitates the identification of a Pareto-complete family of schedulers (which is the goal of Section VI); in the restricted class of static schedulers analysed in [2], [3], it turns out that all policies are efficient.

Aside from [2], [3], the only prior work we are aware of that analyses a part-queueing, part-loss service system is [4]. In this paper, the authors obtain the performance metrics for all classes in closed form, assuming exponential inter-arrival and service times for all classes, under a certain *static* priority scheduling discipline. However, we note that [4] does not attempt to address the *tradeoff* between the performance of the two classes, which is central to the present work.

From an application standpoint, this paper is also related to the considerable literature on sharing the capacity of a cellular system between voice and data traffic; for example, see [5]–[7]. In this line of work, both voice and data classes are treated as lossy, the focus being on characterizing the blocking probability of each class under different (static and dynamic) admission rules. However, to the best of our knowledge, these papers do not analyse the achievable region of performance vectors, or characterize its Pareto frontier.

We also note that there is a well-developed literature on multiclass queueing systems with multiple tolerant classes on a single server (e.g., conservation laws, pioneered by [8]). The achievable region is well understood in such a 'homogeneous' multi-class setting [9], [10]. Interestingly, in this case, it is known that the static and dynamic achievable regions coincide (see [2]), in contrast with the 'heterogeneous' multi-class setting considered here, where we see that the static achievable region is a strict subset of the dynamic achievable region. Moreover, the achievable region in the homogeneous setting is its own Pareto frontier (i.e., all points of the achievable region are efficient) under work conserving policies, also in contrast with the heterogeneous setting considered here.

II. System Description

We consider a single server queueing system with two job classes: eager customers (also denoted as ϵ-customers) have limited patience and demand service within a short span after arrival, whereas tolerant customers (also denoted as τ-customers) can wait in a queue (of infinite capacity) to be served. The τ-customers can be interrupted either partially (i.e., their service rate may be reduced) or completely by ϵ-customers, but not by other τ-customers. Without loss of generality, we assume *a unit server speed.* We assume that ϵ-customers (respectively, τ-customers) arrive according to a Poisson process with rate λ_ϵ (respectively, λ_τ). The sequence of job sizes (a.k.a. service requirements) for both the classes is i.i.d., with B_ϵ denoting a generic ϵ job size, and B_τ denoting a generic τ job size. Throughout, *we assume that B_τ is exponentially distributed with mean $1/\mu_\tau$, and that $\mathbb{E}[B_\epsilon] < \infty$.* Let $\mu_\epsilon := 1/\mathbb{E}[B_\epsilon]$.

A. Dynamic schedulers

We consider dynamic scheduling, wherein the scheduling policy used for the eager class is dependent on the number of tolerant customers in the system (see Footnote 1). The tolerant queue in turn utilizes the service capacity left unused by the eager class in a work-conserving manner. As a result, the service processes of the two classes are interdependent (unlike in the case of static scheduling as considered in [2], [3]). Our dynamic schedulers are of nested type: a top-level policy chooses the sub-policy used for scheduling the ϵ-class based on the occupancy (state) of the τ-class. Consider $\mathbf{g}$ contiguous partitions $\{\mathcal{G}_j\}_{j\leq \mathbf{g}}$ of the non-negative integers. A single sub-policy is used to schedule the ϵ-class[2] when the tolerant queue occupancy lies in $\mathcal{G}_j$ for each $j \leq \mathbf{g}$.

ϵ-schedulers: Note that while the occupancy of the tolerant queue dictates the selection of ϵ sub-policy, the sub-policies are themselves oblivious to the state of the tolerant queue. Moreover, what we refer to as a sub-policy includes system decisions (e.g., service capacity allocated to ϵ-class, admission control, amount of waiting space, etc.) as well as behavioural aspects of the impatient eager customers (e.g., eager customers may balk based on the system occupancy).

We make the following additional assumptions. Some example schedulers that satisfy these are provided in Section II-C:

A.1 To simplify the transition from one ϵ-sub-policy to the next, we assume that all ϵ-customers are dropped when there is an arrival/departure in the τ-queue.[3]

A.2 The scheduling of each sub-policy depends only on the number of ϵ-jobs present in the system.

A.3 Under each sub-policy, there exists a (finite) upper bound on the number of ϵ-jobs in the system at any time.

A.4 Under each sub-policy, the interval between the start of two successive busy periods of the eager class has finite second moment.

τ-schedulers: Next, we state our assumptions on the scheduling policy of the tolerant class.

[2]The further details of ϵ-scheduling policy (after initial selection) will obviously remain constant till the next τ-change, thus these nested schedulers are not restricted, when one considers all possible partitions.

[3]**A**.1 is required for our proof of Theorem 2 (characterizing the blocking probability of the eager class under the SFJ limit). However, under the SFJ limit, this 'flushing' of the ϵ-system is only performed at a bounded rate (since arrivals/departures in the τ-queue occur at a bounded rate), while the arrival rate of the ϵ system scales to infinity. Thus, we expect that this assumption will not impact the blocking probability of the eager class (also evident from the Monte Carlo simulation based study presented in Section VI).

B.1 The τ-scheduler is work conserving, i.e., it utilizes all the service capacity left unused by ϵ-jobs, so long as the τ-queue is non-empty.

B.2 The τ-jobs are served in a serial fashion, i.e., τ-jobs cannot pre-empt one another.

B.3 The τ-scheduler is blind to the size of τ-jobs.

Assumption **B**.1 implies that the tolerant class experiences a time varying service process, which depends on both the τ-state as well as the ϵ-state. Assumptions **B**.2-3 imply that we consider τ-schedulers which are non-pre-emptive and non-anticipative, for instance, *first come first served* (FCFS), *last come first served* (LCFS), and *random order of service* [Chapter 29] [11].

We require another assumption (**B**.4) regarding the stability of the τ-queue under the SFJ limit, when ϵ-customers employ a single sub-policy (irrespective of τ-state). These are referred to as τ-static schedulers in [3]. We provide the required background on these schedulers, define formally the SFJ scaling, and state the resulting pseudo-conservation law (see [3] for more details), after which we state Assumption **B**.4.

B. τ-static schedulers and background

We now consider the special case of τ-static scheduling, where a single ϵ-sub-policy is used at all times (irrespective of τ-state); this case was analysed in [3]. Let P_{Bj} represent the blocking probability (long run fraction of losses) of the ϵ-class, if sub-policy-j is used in a τ-static manner (P_{Bj} was referred to as the standalone blocking probability of sub-policy j in Section I). We call these as τ-static blocking probabilities.

Short-Frequent Jobs (SFJ) Scaling: Under the SFJ scaling (as in [3]), we let $\lambda_\epsilon \to \infty$ and $\mu_\epsilon \to \infty$, such that $\rho_\epsilon := \lambda_\epsilon/\mu_\epsilon$ remains constant. This corresponds to scaling the arrival as well as the service rate of the eager class to infinity proportionately, so that the offered load (the long term rate at which work arrives into the system) is held constant. We use μ_ϵ as the scale parameter for this partial scaling. Specifically, we scale the job size distribution of the eager class as, $B_\epsilon^{\mu_\epsilon} \stackrel{d}{=} B_\epsilon^1/\mu_\epsilon$, where $B_\epsilon^{\mu_\epsilon}$ denotes a generic eager job size at scale μ_ϵ and $\stackrel{d}{=}$ is equality in distribution. This scaling (plus Poisson arrivals) under **A**.2 ensures that the occupancy process of the ϵ-class gets time-scaled (fast-forwarded) by μ_ϵ; see the proof of Theorem 2 in the Appendix for more details. Note that the tolerant workload remains unscaled. Thus, the SFJ scaling may be viewed as a time-scale separation, with the eager class operating at a faster time-scale.

Static Pseudo Conservation: Let $\Omega_j^{\mu_\epsilon}(t)$ represent the total amount of server capacity left unused by the ϵ-customers in time interval $[0,t]$, under sub-policy j operating in a τ-static manner. Note that $\Omega_j^{\mu_\epsilon}(t)$ is the (cumulative) service process seen by the τ-system. Then by [3, Lemma 1], for all μ_ϵ, the asymptotic (in time) growth rate of $\Omega_j^{\mu_\epsilon}(t)$ is the same (a.s.):

$$\lim_{t\to\infty} \frac{\Omega_j^{\mu_\epsilon}(t)}{t} \;\to\; \nu_j \;,\; \nu_j \;:=\; 1-\rho_\epsilon(1-P_{Bj}) \text{ almost surely.} \quad (1)$$

In other words, the long run time average service rate seen by the τ-queue equals ν_j, which depends only on the blocking probability P_{Bj} of the eager class, and not on the specific ϵ-sub-policy that produced the blocking probability. Further under the SFJ limit, the service process seen by the τ-class becomes uniform. Specifically it follows from [3, Theorem 1] that, as $\mu_\epsilon \to \infty$ in the SFJ limit,

$$\sup_{t\le W} \left|\Omega_j^{\mu_\epsilon}(t) - \nu_j t\right| \to 0 \text{ a.s. for any finite } W, \text{ and}$$
$$\Upsilon_j^{\mu_\epsilon} \stackrel{a.s.}{\to} \frac{B_\tau}{\nu_j}, \text{ both for any initial } \epsilon\text{-state,} \quad (2)$$

where $\Upsilon_j^{\mu_\epsilon}$ denotes the time required to finish B_τ amount of work using the service process $\Omega_j^{\mu_\epsilon}(\cdot)$. This uniformity of the service process under SFJ limit enables a closed form characterization of the performance of the τ-class (see [3, Theorems 2,3]). A key feature of the above results is a *pseudo-conservation law* that expresses the performance of the tolerant class purely in terms of the blocking probability of the eager class, independent of the underlying ϵ-policy that produced the blocking probability. We show an analogous pseudo-conservation for dynamic scheduling policies in this paper.

Finally, we state the following assumption, which ensures that the τ-queue remains stable under each of the $\mathbf{g}$ sub-policies, when they are applied in a τ-static manner.

B.4 There exists $\delta > 0$ such that $\rho_j := \frac{\lambda_\tau}{\mu_\tau \nu_j} < 1-\delta$ for all j.

Assumption **B**.4 guarantees that the τ-system is stable in the dynamic setting as well, as shown by Lemma 1.

C. Some example models

We begin with the description of one example system that satisfies our assumptions, in which the system capacity is not completely transferred to one class at any time, but rather a fraction of it is used by each ϵ-customer, whilst the left is utilized by one τ-customer.

Each ϵ-customer uses $(1/K)$ part of the service capacity. If there are $0 \le \ell \le K$ number of ϵ-customers receiving service, then ϵ-customers are served at a net service rate of (ℓ/K), while the τ-customer in service (if any) is served at rate $((K-\ell)/K)$. This continues up to K ϵ-customers, and any further ϵ-arrival departs without service. Note that whenever an existing ϵ-customer departs, *the service rate of the τ-customer gets increased by* $1/K$. Further there is a prior admission control on ϵ-arrivals, they are admitted with probability p independent of all other events. This is the description of the ϵ-sub-policy (as in [2], [3]). Now the top-level policy varies the probability of admission p based on the occupancy of the τ-queue.

We refer to the above ϵ-sub-policy as Capacity Division or briefly as the CD-(p,K) policy. Note that the CD-(p,K) policy captures a multi-server setting for the eager class. While the ϵ-scheduler need not be work conserving, the τ-class uses all the left over capacity. One can have other ϵ-sub-policies either designed by the system and/or influenced by the impatient response of the ϵ-class. We describe a few here.

Limited Processor Sharing (LPS): This sub-policy, denoted by LPS-(p,K) (as in [2], [3]), admits an incoming eager job into the system with probability p, so long as the number of eager jobs already in service is less than or equal to K. The entire service capacity of the server is shared equally between

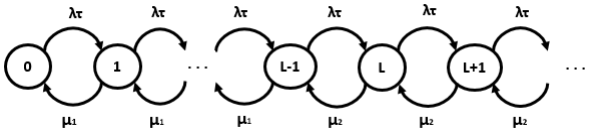

Fig. 1. SDSR-M/M/1 $(\lambda_\tau, 1, \{\mu_1, \mu_2\}, \{\mathcal{G}_1, \mathcal{G}_2\})$ queue: Birth death chain-with $\mathcal{G}_1 = \{0, \cdots, L-1\}$ and $\mathcal{G}_2 = \{L, L+1, \cdots\}$.

the eager jobs in service (i.e., each eager job gets served at rate $1/\ell$ when there are ℓ jobs in service). Note that under the LPS-(p, K) policy, the tolerant class receives service only when there are no eager jobs in the system.

Balking and Reneging: As a part of a particular ϵ-sub-policy, the system may allocate a certain number of servers (recall that the system may be viewed as multi-server from the standpoint of the eager class) and might allocate a certain amount of waiting space for ϵ-class. The ϵ-customers might respond to the resources allocated based on their patience levels: a) an ϵ-customer may not enter the system depending upon the ϵ-number already in system according to some probabilistic rule, as in *balking* models; or b) may leave the system after waiting for an exponentially distributed patience time of rate α^{μ_ϵ}, as in *reneging* models. In the case of reneging, we will require that the parameter α^{μ_ϵ} scales linearly with μ_ϵ, i.e., $\alpha^{\mu_\epsilon} = \alpha\mu_\epsilon$ for some $\alpha \in (0, \infty)$ (as in [3]).

Top-level policies: The top-level policy can chose any one of these sub-policies for any τ-state. For example, when the τ-occupancy is greater than a certain threshold L, one may allocate fewer individual servers to ϵ-customers (using, for example, the CD policy), while one may allocate the entire capacity to the ϵ-class and may serve them in LPS mode when the τ-occupancy is smaller. Alternatively, one may allocate (say) only one server to the ϵ-class when the τ-occupancy is high (or low), which might lead to increased levels of balking/reneging by the ϵ-class.

III. Performance of Tolerant Class

In this section, we characterize the performance of the τ-class under the SFJ limit. Recall that the τ-queue is served (in a work conserving manner) using the unused service process of the ϵ-sub-policy, which in turn is selected based on τ-occupancy. Thus, the τ-queue is served using a random, time varying, state-dependent service process. Due to space constraints, in the present paper, the performance characterization of the tolerant and the eager class is done assuming finitely many ϵ sub-policies, i.e., $\mathbf{g} < \infty$. The generalization to the case of countably many ϵ sub-policies, which is in fact required for our analysis to hold for the class of stationary Markov top-level policies, can be found in [12].

We analyse the τ-performance by considering the τ-queue at arrival/departure epochs. Let X_n denote the occupancy of the τ-queue immediately following the nth arrival/departure. Under Assumption **A**.1 and because of exponentially distributed tolerant job sizes, $\{X_n\}$ is a discrete-time Markov chain with birth-death structure. Our first observation is that this process is positive recurrent for large enough μ_ϵ (proof in Appendix):

Lemma *1:* Assume $\mathbf{g} < \infty$. There exists $\bar{\mu} > 0$ such that for $\mu_\epsilon > \bar{\mu}$, the Markov chain $\{X_n\}$ is positive recurrent. ■

Lemma 1 implies that for large enough μ_ϵ, the τ-queue is stable and has a well defined stationary behaviour.

The performance of the tolerant class under the SFJ limit is characterized in terms of a certain state-dependent service rate M/M/1 (SDSR-M/M/1) queue, which we describe now. An SDSR-M/M/1 queue sees the same workload process as an M/M/1 queue: job arrivals are according to a Poisson process (of rate λ), job sizes are i.i.d. and exponentially distributed (with mean $1/\mu$). However, unlike the standard M/M/1 queue, the SDSR-M/M/1 queue has a state dependent service rate (a.k.a. server speed). Specifically, given a partition $\{\mathcal{G}_j\}_{j \le \mathbf{g}}$ of the non-negative integers, the server operates with service rate ν_j if the number of jobs in the queue (including the job in service) lies in $\mathcal{G}_j$. Thus, the SDSR-M/M/1 queue is parametrized by $(\lambda, \mu, \boldsymbol{\nu}, \{\mathcal{G}_j\}_{j \le \mathbf{g}})$, where $\boldsymbol{\nu} = \{\nu_j\}_{j \le \mathbf{g}}$ is the vector of service rates.

The number of jobs in the SDSR-M/M/1 queue evolves as a continuous time Markov process with birth-death structure (see Figure 1), whose steady state behaviour can be obtained by elementary techniques. In particular, the stationary distribution $\boldsymbol{\pi} := \{\pi(i)\}_{i=0}^{\infty}$, and the expectation of the steady state queue occupancy (denoted by N) are given by (see [13]):

$$\begin{aligned} \pi(i) &= \frac{\mathbf{1}_{\{i=0\}} + \mathbf{1}_{\{i \ge 1\}} \prod_{\ell=1}^{i} \rho_\ell}{1 + \sum_{k \ge 1} \prod_{\ell=1}^{k} \rho_\ell}, \ \rho_\ell := \frac{\lambda}{\mu \nu_j} \text{ if } \ell \in \mathcal{G}_j, \\ E[N] &= \sum_{i=1}^{\infty} i\pi(i). \end{aligned} \tag{3}$$

We are now ready to characterize the performance of the tolerant class under the SFJ limit, for the case with finite number of ϵ-sub-policies (proof in Appendix).

Theorem *1:* **[Number in system]** Assume $\boldsymbol{A}$.1-4 and $\boldsymbol{B}$.1-4. Also assume $\mathbf{g} < \infty$.

i) Under the SFJ scaling, as $\mu_\epsilon \to \infty$, the steady state number of τ-jobs in the system converges in distribution to the steady state number of jobs in an SDSR-M/M/1 $(\lambda_\tau, \mu_\tau, (\nu_1, \nu_2, \cdots \nu_{\mathbf{g}}), \{\mathcal{G}_j\})$ queue, with

$$\nu_j := (1 - \rho_\epsilon(1 - P_{Bj})), \text{ for any } j \le \mathbf{g}.$$

ii) The stationary expected number of τ-customers converges to that of the same limit system (given by (3)). ■

To provide intuition for Theorem 1, note from (2) that $\Upsilon_j^{\mu_\epsilon}$, the time required to complete B_τ amount of job (when uninterrupted) converges to B_τ/ν_j, which is exponentially distributed. Thus one can anticipate the following τ-system at SFJ limit (further because the residual service times are exponentially distributed): a) Poisson arrivals; b) exponential service times, whose rate depends upon the τ-number in the system. And this is precisely the SDSR-M/M/1 queue.

IV. Performance of Eager Class

We now focus on the performance of eager class. As already discussed the ϵ sub-policy changes dynamically among $\mathbf{g}$ sub-policies depending only upon the τ-number in the system (we assume $\mathbf{g} < \infty$ in this section; the generalization to arbitrary partitions is in [12]). To be more precise, if the number of τ-customers at τ-transition (arrival/departure) is in group $\mathcal{G}_j$ of states, sub-policy j is used till the next τ-transition.

The ϵ-class is a lossy system, and the blocking probability would be P_{Bj} if j-th sub-policy is used in τ-static manner. We now derive the 'dynamic' blocking probability, when these sub-policies are selected based on τ-dynamics. We show that, in SFJ limit, the overall blocking probability of the eager class is a convex combination of the τ-static blocking probabilities $\{P_{B_j}\}$, weighted by the long run fractions of time the τ-system spends in the groups $\{\mathcal{G}_j\}$.

Theorem *2:* **[Blocking probability of eager class]** Assume **A**.1-4, **B**.1-4. Also assume $\mathbf{g} < \infty$. Let $\boldsymbol{\pi}_\tau^\infty := \{\pi_\tau^\infty(i)\}_{i\geq 0}$ denote the stationary distribution of SDSR-M/M/1 limit tolerant system, given by Theorem 1. Then the steady state blocking probability of ϵ-jobs in SFJ limit is given by:

$$P_B^{\mu_\epsilon} \quad \overset{\mu_\epsilon \to \infty}{\to} \quad \sum_{j=1}^{\mathbf{g}} P_{Bj} \left(\sum_{i \in \mathcal{G}_j} \pi_\tau^\infty(i) \right) =: P_B^\infty. \qquad \blacksquare$$

A sketch of the proof of Theorem 2 can be found in the appendix; the complete proof, which also generalizes to countably many partitions, can be found in [12].

Dynamic Pseudo Conservation and its relevance: The key challenge in the performance evaluation of our multi-class system is the interdependence between the service processes of the two classes. However, Theorems 1-2 show that one may approximate the performance of both classes (the approximations being accurate under the SFJ limit) using only the τ-static blocking probabilities $\{P_{B_j}\}_{j\leq \mathbf{g}}$. The probabilities $\{P_{B_j}\}_{j\leq \mathbf{g}}$ themselves are typically easy to compute, since they involve the analysis of a single-class (stationary) loss system. Finally, we note that by virtue of Theorems 1-2, we have a *'Dynamic Pseudo Conservation'*: Under the SFJ limit, the performance of *both* classes depends only on the τ-static blocking probabilities $\{P_{B_j}\}_{j\leq \mathbf{g}}$ and the partitions $\{\mathcal{G}_j\}_{j\leq \mathbf{g}}$, and not on other specifics of the $\mathbf{g}$ sub-policies.

V. Dynamic Achievable Region

A queuing system can be analysed using several performance metrics; for example, number of customers in the system, sojourn time (the total time spent by the customer), waiting time (of the customer before the service starts), fraction of the customers blocked (in a loss system), etc. The achievable region of a multi-class system is defined as the region of all possible vectors (one component for one class) of the relevant performance metrics. In our model, corresponding to the eager class we have a lossy system, thus we consider blocking probability as the performance metric. For the tolerant class, one can consider the steady state expected number of customers in the system as the performance metric.

By Lemma 1, the system is stable for all $\mu_\epsilon \geq \bar{\mu}$, for some $\bar{\mu} < \infty$. Thus for all such μ_ϵ, by Little's Law, $E^{\mu_\epsilon}[S]$, the stationary expected sojourn time of a typical τ-customer, and $E^{\mu_\epsilon}[N]$, the stationary expected number of τ-customers in the system are related as $E^{\mu_\epsilon}[N] = \lambda_\tau E^{\mu_\epsilon}[S]$. Thus it is sufficient to consider any one of these metrics.

Stationary Markov top-level policies: In any general sequential decision problem, a Stationary Markov (SM) policy is a sequence of decisions, in which one decision is chosen for each value of the state and the same decision is applicable in any time slot. In our case we consider *the top-level policies among the Stationary Markov (SM) family.* This means, a top level policy ϕ is a sequence of ϵ-sub-policies, and that if the τ-state equals j at any time slot, then the j-th sub-policy of ϕ is used for scheduling the ϵ-class.

While the statements of Theorems 1-2 in the present paper assume finitely many ϵ-sub-policies (one for each of the subsets in $\{\mathcal{G}_j\}$), the study of SM strategies requires us to move on to the general case of countably infinite ϵ-sub-policies, one for each value of τ-occupancy. However, as stated before, the statements of Theorems 1-2 do extend to general SM top-level policies (see [12]). *Accordingly, in the remainder of this paper, we proceed with our analysis of the achievable region and its Pareto frontier disregarding the restriction to finitely many ϵ-sub-policies.*

As understood from Theorems 1-2, the *only characteristic of the ϵ-sub-policies that influences the system (dynamic) performance are the τ-static blocking probabilities* $\{P_{Bj}\}_j$, obtained when respective sub-policies are used in τ-static manner. Thus to define an efficient dynamic system, one effectively needs to choose (based on the τ-state), one among these blocking probabilities (and no further details of the sub-policy are important). This *is a consequence of the 'dynamic pseudo-conservation' mentioned in the previous section.*

Any Stationary Markov (SM) top-level policy is generally given by a sequence of ϵ-sub-policies, one for each τ-state. However, in view of the above observation, a stationary Markov policy can be thought of as a sequence of ϵ-blocking probabilities (derived when the corresponding sub-policies are applied in τ-static manner. In other words, a SM top-level policy is defined by $\phi = (d_0, d_1, \cdots)$, where decision d_j specifies a 'τ-static blocking probability' to be chosen when number of τ-customers equals j. *Towards this we implicitly require the existence of at least one sub-policy, that achieves the given value of 'τ-static blocking probability', which is any value between the system specified limits $\underline{d} := \underline{P_B}$ (minimum possible blocking probability) and $\bar{d} := \overline{P_B}$* (the maximum possible blocking probability). This for example, is achieved by CD-(p, K)/LPS-(p, K) policies mentioned in Section II, when one considers all possible values of $\{(p, K)\}$ (see [2], [3] for more details). In the rest of the paper, we refer to the top-level policies simply as policies for brevity.

Limit Achievable region Our focus from here on will be the dynamic achievable region $\mathcal{A}^\infty$ of performance vectors under the SFJ limit. Recall that for tolerant class, the limit is an SDSR-M/M/1 queue. The ϵ-limit can be seen as a mixture model made up of many lossy systems, each described by their τ-static blocking probabilities, and mixed independently according the stationary distribution of the limit SDSR-M/M/1 queue. Thus, we define the limit achievable region as follows:

$$\mathcal{A}^\infty = \left\{ \left(P_{B,\phi}^\infty, E_\phi^\infty[N] \right) : \phi \text{ is an SM policy} \right\}.$$

Note that $\mathcal{A}^\infty$ is the set of limiting performance vectors under SM policies. In this sense, one may view $\mathcal{A}^\infty$ as the limit of

achievable region of our multi-class system as $\mu_\epsilon \to \infty$.

For simplicity of notations *we avoid the super-script* ∞ when the discussion is clearly about the limit system. *At times we also drop* ϕ*, the SM policy, when there is no ambiguity.*

A Numerical Example: To visualize the limit achievable region, we consider a system with a top-level policy parametrized by $((p_1, p_2, L, K))$. In this system, the CD-(p_1, K) policy is employed when the τ-occupancy is less than L, and the CD-(p_2, K) policy is employed when the τ-occupancy is greater than or equal to L. The tolerant customers are served serially with total capacity of all the leftover servers. Using the Erlang-B formula, the two τ-static blocking probabilities of ϵ-customers equal $\frac{(K\rho_\epsilon p_i)^K}{K!}$

$$\begin{aligned} P_{B\,i} &= (1-p_i) + p_i \frac{\frac{(K\rho_\epsilon p_i)^K}{K!}}{\sum_{k=0}^{K} \frac{(K\rho_\epsilon p_i)^k}{k!}}, \text{ for } i = 1, 2 \text{ and} \\ \mathcal{G}_1 &= \{0, \cdots, L-1\} \text{ and } \mathcal{G}_2 = \{L, L+1, \cdots, \}. \end{aligned}$$

The performance of such a system at limit can be obtained using the results of Theorems 1-2. We set $K = 5$, $\rho_\epsilon = 0.4$, $\lambda_\tau = 4$ and $\mu_\tau = 8$, generate the three parameters (p_1, p_2, L) randomly. The scatter plot of the corresponding values of $E^\infty[N]$ and P_B^∞ is shown in Figure 2. The resulting figure is a part of the limit achievable region. As seen from the figure, the achievable region is a non-zero measure set. Further, the plot indicates that the achievable region is bounded. We will now address the Pareto frontier associated with this system.

VI. Limit Pareto frontier

The Pareto frontier is the efficient sub-region of an achievable region which consists of dominating performance vectors. A pair $(P_{B,\phi^*}, E_{\phi^*}[N])$ (produced by a policy ϕ^*) is on Pareto frontier of the limit system, if there exists no other SM policy ϕ that achieves a better performance pair $(P_{B,\phi}, E_\phi[N])$ (in the limit system), i.e., $P_{B,\phi} \le P_{B,\phi^*}$ and $E_\phi[N] \le E_{\phi^*}[N]$, one of the inequalities being strict.

The Pareto frontier of the limit system is obtained by solving an appropriate set of parametrized optimization problems. Prior to that, we discuss the limit performance of both the sub-systems, under any given SM policy. Invoking Theorems 1 and 2 (specifically, their generalizations in [12]), under any ϕ,

$$E_\phi^{\mu_\epsilon}[N] \overset{\mu_\epsilon \to \infty}{\longrightarrow} E_\phi^\infty[N] = \sum_{i=1}^{\infty} i \frac{\pi_i^\phi}{1 + \sum_{l \ge 1} \pi_l^\phi}, \quad (4)$$

$$P_{B,\phi}^{\mu_\epsilon} \overset{\mu_\epsilon \to \infty}{\longrightarrow} P_{B,\phi}^\infty = \frac{d_0}{1 + \sum_{i \ge 1} \pi_i^\phi} + \sum_{i=1}^{\infty} d_i \frac{\pi_i^\phi}{1 + \sum_{l \ge 1} \pi_l^\phi}, \quad (5)$$

$$\pi_j^\phi = \mathbf{1}_{\{j=0\}} + \mathbf{1}_{\{j>0\}} \prod_{i=1}^{j} \rho_i^\phi, \text{ with } \rho_i^\phi := \frac{\lambda_\tau}{\mu_\tau(1 - \rho_\epsilon(1 - d_i))}. \quad (6)$$

A. Pareto-complete family

We now derive a family of *Pareto-complete* policies, i.e., a parametrized family of policies that span the entire Pareto-frontier of our system. One can obtain all the points on the Pareto frontier by considering the following parametrized (by C) constrained optimization problems (with π_i^ϕ defined in (6)).

$$\min_\phi P_{B,\phi} \text{ such that } E_\phi[N] \le C, \text{ i.e., equivalently} \quad (7)$$

$$\min_\phi \sum_{i=0}^{\infty} d_i \frac{\pi_i^\phi}{1 + \sum_{l \ge 1} \pi_l^\phi} \text{ such that } \sum_{i=0}^{\infty} i \frac{\pi_i^\phi}{1 + \sum_{l \ge 1} \pi_l^\phi} \le C.$$

Recall $\underline{d}$, $\bar{d}$ respectively represent the best and worst sub-policy (with respect to ϵ-customers), in that these represent the minimum and maximum possible blocking probabilities. Define:

$$\bar{\rho} := \frac{\lambda_\tau}{\mu_\tau(1 - \rho_\epsilon(1 - \underline{d}))}, \text{ and } \underline{\rho} = \frac{\lambda_\tau}{\mu_\tau(1 - \rho_\epsilon(1 - \bar{d}))}. \quad (8)$$

The terms $(\bar{\rho}, \underline{\rho})$ represent the (worst and best) load factor of the τ-customers in the limit system, when ϵ-customers are scheduled respectively with the best (blocking probability $\underline{d}$) and worst (blocking probability $\bar{d}$) sub-policies, in τ-static manner.

Suppose that the constraint C on the expected τ-number satisfies $C \ge \bar{\rho}/(1-\bar{\rho})$ (observe $\bar{\rho}/(1-\bar{\rho})$ is the expected number in M/M/1 queue with maximum load factor $\bar{\rho}$). Then the problem (7) becomes an unconstrained problem and the optimal policy clearly equals $\phi^* = (\underline{d}, \underline{d}, \cdots)$. We show that the optimal policy for any given $C < \bar{\rho}/(1-\bar{\rho})$, is monotone (but not strictly monotone) in τ-state and further derive its closed form expression (proof in Appendix):

Theorem *3:* The policy $\phi^* = \{d_0^*, d_1^*, \cdots\}$ that optimizes the problem defined in (7) is monotone and is given by:

$$d_i^* = \mathbf{1}_{\{i < L^*\}} \underline{d} + \mathbf{1}_{\{i = L^*\}} d^* + \mathbf{1}_{\{i > L^*\}} \bar{d} \quad (9)$$

which is parametrized by two parameters (L^*, d^*). The expressions for (L^*, d^*) are given in [12]. ■

The family of schedulers given by (9) are clearly Pareto-complete. This family is parametrized by (L, d) with $1 \le L \le \infty$ and $\underline{d} \le d \le \bar{d}$. The policies in this family choose the 'worst' ϵ-sub-policy (i.e., with $d = \bar{d}$) when the τ-number is greater than or equal to $L+1$, choose a sub-policy with intermediate blocking d when τ-number equals L and choose the 'best' sub-policy (i.e., with $d = \underline{d}$) for the rest (see (9)). One can easily compute the performance under these policies, as below (see 8):

$$\begin{aligned} E_{(L,d)}[N] &= \psi\left(\bar{\rho}\frac{1-\bar{\rho}^L}{(1-\bar{\rho})^2} - \frac{L\bar{\rho}^{L-1}}{1-\bar{\rho}} + \frac{\rho\bar{\rho}^{L-1}(\underline{\rho} + L - L\underline{\rho})}{(1-\underline{\rho})^2}\right) \\ P_{B,(L,d)} &= \psi\left(d\rho\bar{\rho}^{L-1} + \rho\bar{d}\frac{\bar{\rho}^{L-1}\underline{\rho}}{1-\underline{\rho}} + \frac{\bar{\rho}^L - 1}{\bar{\rho} - 1}\underline{d}\right) \text{ with} \\ \psi &= \frac{1}{\frac{\bar{\rho}^L - 1}{\bar{\rho} - 1} + \rho\bar{\rho}^{L-1} + \rho\frac{\bar{\rho}^{L-1}\underline{\rho}}{1-\underline{\rho}}}, \quad \rho = \frac{\lambda_\tau}{\mu_\tau(1 - \rho_\epsilon(1-d))} \quad . \quad (10) \end{aligned}$$

Thus *we derived Pareto complete family as well as the performance under this family, which can readily be used for any relevant optimization problem.*

Numerical example: We continue with the numerical example of Figure 2. For this example, one can easily compute that $\bar{\rho} = 0.8134$ (no admission control on eager class, i.e., with $p_i = 1$ for all i) and $\underline{\rho} = 0.5$ (eager class is completely blocked with $\bar{d} = 1$ and hence $\underline{\rho} = \lambda_\tau/\mu_\tau$), when the system can at maximum serve 5 eager customers in parallel. By substituting these values into (10), one can obtain the Pareto frontier. The circles in the figure represent this Pareto frontier, and are obtained by varying (L, d) appropriately. It is clear from the figure that the derived set of points are indeed dominating and are on the Pareto frontier.

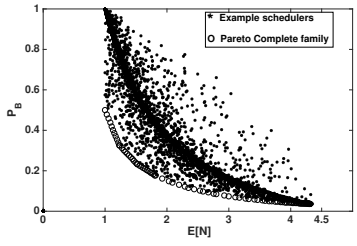

Fig. 2. Achievable region, Pareto Frontier $\bar{\rho} = 0.8134$, $\underline{\rho} = .5$

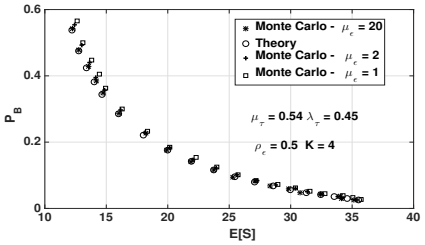

Fig. 3. Comparison of Theory with Monte Carlo (MC) Estimates. Good approximation even for small μ_ϵ.

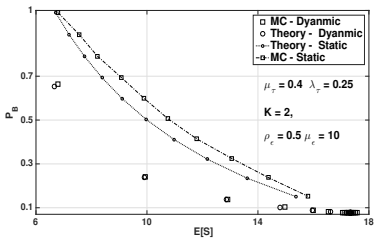

Fig. 4. Comparison of static and dynamic policies. Better approximation for dynamic policies.

B. Monte Carlo based case study in pre-limit

We consider an example case-study with CD-(p, K) sub-policies of Subsection II-C. Specifically, for fixed K, we consider top-level policies that perform CD-$(1, K)$ when the τ-occupancy is less than certain L (with $L \geq 1$) and perform CD-$(0, K)$ (thus blocking all ϵ-jobs) when the τ-occupancy is greater than or equal to L. In view of Theorem 3, by stepping over L as above[4], we sample performance vectors from the limit Pareto-frontier of the system.

It is very complicated to obtain an exact analysis of this heterogeneous system. However by Theorems 1-2, one can obtain an approximate analysis for this system and the same is plotted in Figure 3 (the system configuration is mentioned in the figure itself). We plot both Monte-Carlo estimates as well as the corresponding theoretical limits for different values of L. Importantly, *the Monte Carlo simulations do not even drop ϵ-customers at τ-transitions as required by* **A**.*1*. We observe that Theorems 1-2 provide an excellent approximation for the performance of actual system even for μ_ϵ as small as 1, when $\mu_\tau = 0.54$. It is also apparent that the theory approximates the system performance well even when the system does not drop ϵ-customers at τ-transitions.

Another example is plotted in Figure 4, where τ-static policies of [3] are considered along with dynamic policies. We again observe a good approximation between the theory and MC estimates for dynamic policies. Interestingly, the approximation error is bigger in the static case. One possible explanation for this is the following. It is clear that the approximation error gets smaller as the ϵ-load factor reduces, under τ-static policies. Under Pareto optimal family of schedulers, the ϵ-load equals 0 for all τ-states greater than L. Thus we see the approximation is almost zero towards the right of the two figures (as P_B gets smaller, L gets smaller). We also observe that the dynamic policies perform far superior than τ-static policies.

VII. Concluding remarks

In this paper, we analyse a multi-class, single server queueing system with an eager (lossy) class and a tolerant (queueing) class, under dynamic scheduling. While the inter-dependence between the service processes of the two classes makes an exact analysis of this system difficult, we obtain tractable performance approximations under a certain (partial) fluid scaling regime. A key feature of our approximations, which are shown to be highly accurate via Monte Carlo simulations, is a pseudo-conservation law: the approximate performance of both classes is expressed in terms of the standalone blocking probabilities of the eager schedulers, which are themselves easy to compute in several cases. Finally, we focus on the achievable region of the limiting performance vectors for our system. Remarkably, we are able to obtain an explicit family of Pareto-optimal policies (these resemble threshold policies).

This work motivates extensions in various directions. One interesting extension would be to the multi-server setting, where the tolerant class is no longer work conserving. Another promising direction is to consider static/dynamic pricing for such heterogeneous service systems. Finally, specializing our models to particular application scenarios, including supermarkets, cognitive radio, and cloud computing environments, would be of independent interest.

[4]Here again, $\underline{d}$ (respectively $\bar{d}$) equals the blocking probability without eager admission control (respectively if eager class is admitted only when τ-queue is empty).

References

[1] S. R. Mahabhashyam and N. Gautam, "On queues with markov modulated service rates," *Queueing Systems*, vol. 51, no. 1-2, pp. 89–113, 2005.

[2] Veeraruna Kavitha and R. K. Sinha, "Achievable region with impatient customers," in *Proceedings of Valuetools*, 2017.

[3] Veeraruna Kavitha, J. Nair, and R. K. Sinha, "Pseudo conservation for partially fluid, partially lossy queueing systems," *Annals of Operations Research*, pp. 1–38, 2018.

[4] A. Sleptchenko, A. van Harten, and M. C. van der Heijden, "An exact analysis of the multi-class m/m/k priority queue with partial blocking," *Stochastic models*, vol. 19, no. 4, pp. 527–548, 2003.

[5] B. Li, L. Li, B. Li, K. M. Sivalingam, and X.-R. Cao, "Call admission control for voice/data integrated cellular networks: performance analysis and comparative study," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 4, pp. 706–718, 2004.

[6] S. Tang and W. Li, "A channel allocation model with preemptive priority for integrated voice/data mobile networks," in *Proceedings of the First International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks*, 2004.

[7] Y. Zhang, B.-H. Soong, and M. Ma, "A dynamic channel assignment scheme for voice/data integration in gprs networks," *Computer Communications*, vol. 29, no. 8, pp. 1163–1173, 2006.

[8] L. Kleinrock, "A conservation law for a wide class of queueing disciplines," *Naval Research Logistics Quarterly*, vol. 12, no. 2, pp. 181–192, 1965.

[9] E. G. Coffman Jr and I. Mitrani, "A characterization of waiting time performance realizable by single-server queues," *Operations Research*, vol. 28, no. 3-part-ii, pp. 810–821, 1980.

[10] J. G. Shanthikumar and D. D. Yao, "Multiclass queueing systems: Polymatroidal structure and optimal scheduling control," *Operations Research*, vol. 40, no. 3-supplement-2, pp. S293–S299, 1992.

[11] M. Harchol-Balter, *Performance modeling and design of computer systems: Queueing theory in action*. Cambridge University Press, 2013.

[12] Kiran Chaudhary, Veeraruna Kavitha, and J. Nair, "Dynamic scheduling in a partially fluid, partially lossy queueing system," *ArXiv e-prints, https://arxiv.org/abs/1904.06480* , 2019.

[13] R. G. Gallager, *Discrete stochastic processes*. Springer Science & Business Media, 2012.

APPENDIX

Proof of Lemma 1: Pick $i \geq 1$ such that $i \in \mathcal{G}_j$. Define $p_j^{\mu_\epsilon} := P\left(X_{n+1} = i+1 \mid X_n = i\right)$. To show that the birth-death chain, $\{X_n\}$ is positive recurrent for large enough μ_ϵ, it suffices to show that (using standard Lyaponov arguments)

$$\lim_{\mu_\epsilon \to \infty} p_j^{\mu_\epsilon} = \frac{\lambda_\tau}{\lambda_\tau + \mu_\tau \nu_j} < \frac{1}{2}, \tag{11}$$

where the final inequality is a consequence of Assumption **B**.4. Indeed, given the assumption of finitely many eager sub-policies, (11) implies that for a suitably small $\delta > 0$, there exists $\bar{\mu} > 0$ such that for $\mu_\epsilon > \bar{\mu}$, $p_j^{\mu_\epsilon} < 1/2 - \delta$ for all j. To prove (11), note that

$$p_j^{\mu_\epsilon} = P\left(A_\tau < \Upsilon_j^{\mu_\epsilon}(B_\tau)\right) = 1 - \mathbb{E}\left[\exp(-\lambda_\tau \Upsilon_j^{\mu_\epsilon}(B_\tau))\right],$$

where $\Upsilon_j^{\mu_\epsilon}(B_\tau)$ denotes the time required to accumulate a service of B_τ using the unused service process of the eager class under sub-policy j, starting with an empty eager queue. Since $\Upsilon_j^{\mu_\epsilon}(B_\tau) \to B_\tau/\nu_j$ a.s. as $\mu_\epsilon \to \infty$ (by [3, Theorem 1], and, see (2)), and since $\exp(-\lambda_\tau \Upsilon_j^{\mu_\epsilon}(B_\tau)) \leq 1$, it follows from the bounded convergence theorem that $\lim_{\mu_\epsilon \to \infty} p_j^{\mu_\epsilon} = 1 - \frac{\mu_\tau}{\mu_\tau + \lambda_\tau/\nu_j}$. ■

Proof of Theorem 1 Let $X(t)$ denote the τ-queue occupancy at time t. Under Assumption **A**.1, $\{X(t)\}$ is a semi-Markov process, with $\{X_n\}$ being its embedded Markov chain (EMC). Moreover, since the mean time spent in each state of the semi-Markov process is bounded from above ($1/\lambda_\tau$ is a trivial sub-policy independent bound), it follows that the time average distribution of $\{X(t)\}$ is well defined (see [13]).

The first goal is to show that the stationary distribution of $\{X(t)\}$ converges to that of the limit SDSR-M/M/1 queue as $\mu_\epsilon \to \infty$. By Lemma 2 (which establishes a bijection between the stationary distributions of each of the above queues and the stationary distributions associated with the corresponding EMCs), it suffices to show that the stationary distribution of $\{X_n\}$ converges to that of the EMC of the limit SDSR-M/M/1 queue. However, this follows from the convergence of the transition probabilities of the $\{X_n\}$ process to those of the EMC of the limit SDSR-M/M/1 queue (shown in the proof of Lemma 1). Indeed, note that the stationary probabilities are a continuous function of the finite vector $\{p_j^{\mu_\epsilon},\ j \leq \mathbf{g}\}$. The convergence of $\mathbb{E}\left[N^{\mu_\epsilon}\right]$ to $\mathbb{E}\left[N^{\text{SDSR}-\text{M}/\text{M}/1}\right]$ as $\mu_\epsilon \to \infty$ also follows, given that $\mathbb{E}\left[N^{\mu_\epsilon}\right]$ is a continuous function of finite vector $\{p_j^{\mu_\epsilon},\ j \leq \mathbf{g}\}$. ■

Lemma *2:* Consider a stable queueing system with Poisson job arrivals and no simultaneous departures (i.e., jobs depart one at a time with probability 1), such that the time-average distribution of queue occupancy $\pi = \{\pi_i\}_{i \geq 0}$ is well defined, i.e.,

$$\pi_i = \lim_{t \to \infty} \frac{1}{t} \int_0^t \mathbf{1}_{\{X(s)=i\}} ds \quad \forall\ i \quad (a.s.),$$

where $X(t)$ denote the queue occupancy at time t. Let $\tilde{\pi} = \{\tilde{\pi}_i\}_{i \geq 1}$ denote the (discrete-time) time-average distribution of the queue occupancy sampled just following arrival/departure epochs. Then

$$\tilde{\pi}_0 = \frac{1}{2}\pi_0, \text{ and } \tilde{\pi}_i = \frac{1}{2}(\pi_{i-1} + \pi_i) \quad (i \geq 1).$$

The proof is available in [12].

Sketch of proof of Theorem 2: For the purpose of almost sure comparison we construct an ϵ-process for any μ_ϵ and any scheme j (when used in τ-static manner), in a similar way as in [3]. By this construction, the full ϵ-cycles $\{\mathcal{B}^{\mu_\epsilon}\}$ (time duration between start of an ϵ-idle period and the end of the consequent ϵ-busy period) and $\{\Upsilon^{\mu_\epsilon}\}$ (server capacity available to τ-customers during one full ϵ-cycle), for any j, can be compared for various μ_ϵ as below (see [12])

$$\mathcal{B}_j^{\mu_\epsilon} = \frac{\mathcal{B}_j^1}{\mu_\epsilon},\ \Upsilon_j^{\mu_\epsilon} = \frac{\Upsilon_j^1}{\mu_\epsilon},\ N_A^\epsilon(\mathcal{B}_j^{\mu_\epsilon}) = N_A^1(\mathcal{B}_j^1), N_B^\epsilon(\mathcal{B}_j^{\mu_\epsilon}) = N_B^1(\mathcal{B}_j^1)$$

etc., where $N_A^\epsilon(\cdot)$, $N_B^\epsilon(\cdot)$ respectively represent the number of ϵ-arrivals and number of ϵ-drops in the specified time interval. It is clear that the length of an ϵ-full cycle (and Υ^{μ_ϵ}) decreases to zero, while the ϵ-number arrived/served in a full cycle remains the same, as $\mu_\epsilon \to \infty$. This forms the main step in deriving the limits required for this proof. The overall blocking probability can be split as

$$P_B = \lim_{t \to \infty} \frac{\sum_{j=1}^{\mathbf{g}} \left(N_B^\epsilon(\Lambda_j(t)) + N_B^\epsilon(\Psi_j(t))\right) + N_\partial(t)}{N_A^\epsilon([0,t])},$$

where a) $\Lambda_j(t)$ represents the total time consisting of full ϵ-cycles during time period $[0,t]$, such that τ-state is in group $\mathcal{G}_j$; b) $\Psi_j(t)$ is the remaining period (of partial ϵ-cycles) during which τ state is in $\mathcal{G}_j$ before t; and c) $N_\partial(t)$ is the number of ϵ-jobs dropped at τ-transition epochs (see **A**.1).

The $\Lambda_j(t)$ is shown to form a renewal process of special full ϵ-cycles. Hence using Renewal Reward Theorem (RRT) and further using τ-stability as given by Lemma 1 (fraction of time spent in $\mathcal{G}_j$ converges to its stationary probability) we prove that (see [12] for details):

$$\lim_{\mu_\epsilon \to \infty} \lim_{t \to \infty} \frac{N_B^\epsilon(\Lambda_j(t))}{N_A^\epsilon(t)} = P_{B\,j} \pi_\tau^\infty(j) \text{ a.s., for any } j.$$

An upper bound on $\Psi_j(t)$ (the partial cycles bounded by full ϵ-cycles), forms another renewal process. This process has similar number of ϵ-cycles per renewal period, for all μ_ϵ (unlike in $\Lambda_j(t)$), hence become insignificant (almost surely):

$$\lim_{\mu_\epsilon \to \infty} \lim_{t \to \infty} \frac{N_B^\epsilon(\Psi_j(t))}{N_A^\epsilon(t)} = 0 \text{ and } \lim_{\mu_\epsilon \to \infty} \frac{N_\partial(t)}{N_A^{\mu_\epsilon}(t)} \leq \lim_{\mu_\epsilon \to \infty} \frac{2\lambda_\tau K}{\mu_\epsilon} = 0,$$

with K = maximum waiting space (see assumption **A**.3). ■

Sketch of the proof of Theorem 3: We discuss the major steps of the proof, while the details are in [12].

a) We first show that the problem is equivalent to the following modified optimizing problem, which optimizes over the sequence of τ-load factors $\{\rho_i\}$:

$$\max_{\phi = \{\rho_i\}: \underline{\rho} \leq \rho_i \leq \bar{\rho}\ \forall i} \sum_i \prod_{j=0}^{i} \rho_j, \text{ such that } \sum_i (i - C) \prod_{j=0}^{i} \rho_j \leq 0.$$

b) The constraint is satisfied with equality at the optimizer;

c) We then prove the following special 'fullness at initial states' property: If $\phi = (\rho_0, \rho_1 \cdots)$ is any policy such that there exists an i with $\rho_i < \bar{\rho}$ and $\rho_{i+1} > \underline{\rho}$ (i.e., if initial decisions have scope to improve and later decision have scope to lose), then one can construct another policy (by appropriate partial swapping) that strictly improves upon it. ■