Dynamic Scheduling of Multiclass Many-server Queues with Abandonment: the Generalized $c\mu/h$ Rule

Zhenghua Long[†], Nahum Shimkin[‡], Hailun Zhang[†], Jiheng Zhang[†]

[†]Department of Industrial Engineering & Decision Analytics,
The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
zlong@connect.ust.hk, hzhangaq@connect.ust.hk, jiheng@ust.hk

[‡]Department of Electrical Engineering, Technion – Israel Institute of Technology, Haifa 32000, Israel
shimkin@ee.technion.ac.il

We consider the problem of server scheduling in a multiclass many-server queueing system with abandonment. For the purpose of minimizing the long-run average queue length costs and abandon penalties, we propose three scheduling policies to cope with any general cost functions and general patience time distributions. First, we introduce the target-allocation policy, which assigns higher priority to customer classes with larger deviation from the desired allocation of the service capacity, and prove its optimality for any general queue length cost functions and patience time distributions. The $Gc\mu/h$ rule, which extends the well-known $Gc\mu$ rule by taking abandonment into account, is shown to be optimal for the case of convex queue length costs and nonincreasing hazard rates of patience. For the case of concave queue length costs but nondecreasing hazard rates of patience, it is optimal to apply a fixed priority policy, and a knapsack-like problem is developed to determine the optimal priority order efficiently. As a motivating example of the operations of emergency departments, a hybrid of the $Gc\mu/h$ rule and the fixed priority policy is suggested to reduce crowding and queue abandonment. Numerical experiments show that this hybrid policy performs satisfactorily.

Key words: multiclass queue, abandonment, fluid model, dynamic scheduling, knapsack problem

1. Introduction

In 2011, the number of left-without-being-seen (LWBS) patients in the United States was 2.6 million (The National Hospital Ambulatory Medical Care Survey, NHAMCS) for the most common reason of being "fed up with waiting" (Rowe et al. (2006)). Patient crowding in the emergency department (ED) has become an increasing public health problem for hospitals around the world as it contributes to increased LWBS rates and dissatisfaction with care (Pines et al. (2011)). We consider the problem of scheduling triage patients from the waiting room to treatment rooms to reduce ED crowding and LWBS rates.

Upon arrival, patients are rapidly sorted into five triage classes by experienced triage nurses using the Emergency Severity Index (ESI). The acuity levels from level 1 (most critical) to 5 (least critical) are based on patient acuity and resource needs (Gilboy et al. (2011)). The ESI may or may not lead to improved patient flow through the ED since the physician response times for levels 1 and 2 are within minutes, but leaves the majority of lower acuity patients waiting to be called for service according to their triage levels. Many patients visiting EDs are in low-acuity conditions. These patients have limited patience and may abandon the ED before receiving treatment. A new empirical study (Batt and Terwiesch (2015)) indicates that the proportion of patients who abandon is up to 6.5% and this rate ranges from 1.5% to 9.0% for different triage levels. The fundamental question that ED physicians face on a daily basis is: which patient should be called for service first when a treatment bed becomes available? This also gives us a motivating example for treating a general queueing control problem—scheduling of multiclass many-server queues with abandonment.

Recent studies on this scheduling problem have introduced a handy policy, namely the $c\mu/\theta$ rule. This fixed priority scheduling policy has been proved to be asymptotically optimal (Atar et al. (2008, 2010, 2011, 2014)) for linear costs and exponential patience. It is consistent with the ESI system in the sense that high-acuity patients receive high priority. However, this rough treatment ignores the real-time status of the ED system and may lead to long waiting times and high LWBS rates for low-acuity patients. Indeed, the well-known generalized $c\mu$ rule ($Gc\mu$) assigns dynamic priority to the flows of multiple classes of customers (van Mieghem (1995), Mandelbaum and Stolyar (2004), Gurvich and Whitt (2009b)). Recently, this scheduling policy has been applied in the control of patient flows in EDs with feedback (Huang et al. (2015)). However, the $Gc\mu$ rule does not consider the LWBS patients. In this paper, we take into account patience time (the amount of time a patient is willing to wait for service) following general distributions. A natural paradigm to study the ED dynamics would be a multiclass many-server queueing system with abandonment (the LWBS phenomenon) as shown in Figure 1. One of our main results is to introduce a dynamic scheduling policy, which we refer to as the generalized $c\mu/h$ rule ($Gc\mu/h$), to minimize the long-run average queueing costs and abandon penalties.

To describe our $Gc\mu/h$ rule, let μ_i be the service rate of level i patients and F_i denote the patience time distribution of level i patients with the hazard rate function h_i. Denote the marginal queue length cost function and the penalty for each abandonment of level i by $c_i(\cdot)$ and γ_i, respectively. The arrival rates λ_i's are determined by triage nurses when

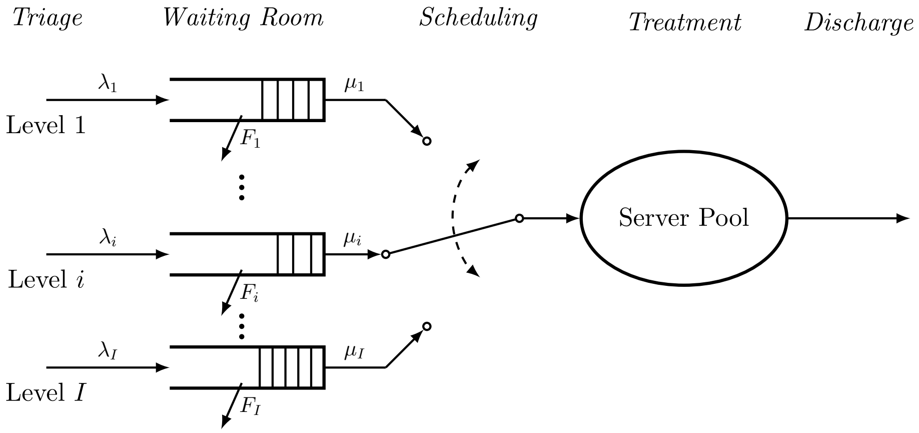

Figure 1 The scheduling problem in EDs with LWBS patients

categorizing ED visits. Let $B_i(t)$ be the number of level i patients being served in the treatment rooms. We call the scheduling policy that serves the level i patient (FCFS within each level) with the highest index

$$i \in \arg\max_i \left(\frac{c_i\big(\lambda_i \int_0^{F_i^{-1}(1-B_i(t)\mu_i/\lambda_i)} F_i^c(s)ds\big)\mu_i}{h_i(F_i^{-1}(1-B_i(t)\mu_i/\lambda_i))} + \gamma_i\mu_i \right),$$

the *generalized $c\mu/h$ rule* ($Gc\mu/h$). We show that the $Gc\mu/h$ rule is asymptotically optimal in a many-server fluid regime with convex queueing costs and nonincreasing hazard rates.

The $Gc\mu/h$ rule can be brought into play not only in call centers but also in systems like EDs due to its flexibility. For call center operations, the latest information technology allows all agents and supervisors to observe the real-time status of the system (Gans et al. (2003)). However, the situation in EDs is quite different. The queue status is usually unknown to ED staff since they are not notified when patients quit waiting. Our scheduling decision suitably depends on the current number of patients in the treatment room. There is no need to modify the rule when the service capacity in the hospital changes. For example, the ED beds may be temporarily added to increase available capacity when all licensed beds are occupied (Derlet et al. (2014)). In such a situation, the $Gc\mu/h$ rule adapts automatically to the change in service capacity.

Our $Gc\mu/h$ rule and the family of $Gc\mu$ rules (van Mieghem (1995), Mandelbaum and Stolyar (2004)) all consider convex queue length costs, but a theoretical understanding of more general cost functions is still lacking. To tackle this problem, we propose another dynamic scheduling policy referred to as the target-allocation policy (see §3.1). In an overcrowded ED, where a portion of the patients may end up leaving without being treated, the number of patients will be stable. The steady state of all types of patients in the treatment rooms can be viewed as an allocation of the service capacity. Our target-allocation policy aims to assign higher priority to the class of patients that deviates most from the optimal allocation, which is determined by solving a nonlinear optimization problem (13). The advantage of this policy is that it is optimal for any general cost functions and patience distributions. However, the primary challenge lies in solving the nonlinear programming in advance.

The current practice in the EDs is mainly to implement triage priority (Batt and Terwiesch (2015)), although the $Gc\mu/h$ rule suggests that the fixed priority rule could be sub-optimal for convex queue length costs and nonincreasing hazard rate functions. Unexpectedly, for concave queue length cost functions and nondecreasing hazard rate functions of patience, we find that the optimal scheduling is a fixed priority policy. It is NP-hard to determine an optimal priority order since it involves the minimization of a concave function. As it is nontrivial to solve a concave optimization problem using standard non-linear approaches, we formulate it as a knapsack-like problem and develop a dynamic programming algorithm. The algorithm can efficiently determine the treatment priority, especially when patients are further categorized by disease types. Our algorithm reduces the time complexity in a similar problem studied in Burke et al. (2008) (see Remark EC.1). The novel research allows us to choose the most appropriate scheduling policy under any queue length cost functions and patience distributions.

The $Gc\mu/h$ rule and the fixed priority rule have their own merits in the sense that the former gives consideration to the least critical patients while the latter enables the most critical patients to receive timely treatment. In view of the fact that the most critical patients may not survive if they fail to receive medical care in time, there is no doubt that they should be given the highest priority. On the other hand, the majority of patients in low-acuity conditions should also be taken care of in a timely manner as they are the main reason for ED crowding and high LWBS rates. To balance the tradeoff, we suggest a hybrid policy to improve patient flows in EDs as follows: according to ESI assign the highest priority to level 1, the second highest priority to level 2, and apply the $Gc\mu/h$ rule to levels 3, 4, and 5. Numerical experiments in §4 show that this hybrid policy achieves the desired allocation of service capacity in the long run.

1.1. Literature Review.

Fluid approximations for many-server queues with general patience time distributions began to emerge following the pioneering work of Whitt (2006). Bassamboo and Randhawa (2010) established the optimal gap of fluid approximation as the system size increases. As an example of how powerful the fluid model approach is that it can be used to approximate a system with dependent service and patience times, see Bassamboo and Randhawa (2016); Wu et al. (2017). For multiclass queues, Atar et al. (2014) established the fluid limit of a multiclass $G/GI/n+GI$ queueing system building on the approach developed by Kaspi and Ramanan (2011). Our fluid model is tailored to a multiclass $G/M/n+GI$ system with exponential service time distributions.

The $c\mu$-type rules have a long history in the study of scheduling problems. As early as Smith (1956) and Cox and Smith (1961), the $c\mu$ rule was proposed and proved to be optimal for a multiclass $M/G/1$ system with linear holding costs. Recently, in Atar et al. (2008, 2010, 2011, 2014), it was extended to the $c\mu/\theta$ rule in the case of exponential abandonment. The $Gc\mu$ rule of van Mieghem (1995) appears to be the first to consider nonlinear, convex holding costs in the analysis of a multiclass $G/G/1$ queue. Mandelbaum and Stolyar (2004) generalized the $Gc\mu$ rule to a system with heterogeneous servers. Our $Gc\mu/h$ rule extends van Mieghem (1995) and Atar et al. (2008, 2010, 2011, 2014) to a multiclass many-server queueing system with general patience and nonlinear holding costs.

Other than the $c\mu$-type rules, there has also been an expanding body of literature on the optimal control of multiclass queueing systems. Harrison and López (1999) explicitly solved a dynamic control problem in the multiclass parallel-server setting. Based on the conventional heavy traffic regime, Ata and Tongarlak (2013) and Kim and Ward (2013) considered dynamic policies by studying the approximating Brownian control problems. Focusing on the Halfin-Whitt scaling proposed by Halfin and Whitt (1981) in the quality-and-efficiency-driven regime, Atar et al. (2004), Atar (2005) and Ata et al. (2012) studied dynamic scheduling policies by formulating a Hamilton-Jacobi-Bellman equation based on the heavy traffic limits; Dai and Tezcan (2008) developed robust control policies to minimize the total linear holding and abandon costs for a parallel server system; Gurvich and Whitt (2009a,b, 2010) studied the staffing and control problems of service systems with multiple customers classes and multiple agent pools; and Kim et al. (2018) solved a diffusion control problem to propose a scheduling policy for a critically loaded multiclass system with abandonment.

1.2. Contributions

The main contributions of this paper are summarized as follows:

- We propose scheduling policies to optimally control a multiclass many-server queueing system under any given queue length cost functions and patience distributions.
- The target-allocation policy is optimal for any general queue length cost functions and patience time distributions by assigning higher priority to customer classes that deviate most from the desired allocation of the service capacity.
- The $Gc\mu/h$ rule extends the $Gc\mu$ rule of van Mieghem (1995) to overloaded systems with impatient customers and is shown to be optimal for convex queue length cost functions and nonincreasing hazard rates of patience.
- The fixed priority policy is proved to be optimal for concave queue length cost functions and nondecreasing hazard rates of patience. It represents a generalization of the $c\mu/\theta$ rule of Atar et al. (2008, 2010, 2011, 2014), which considers linear cost and exponential patience.

The remainder of this paper is organized as follows. In §2, we introduce the fluid model of a multiclass many-server queueing system with abandonment. We also study a steady-state optimization problem. Our proposed policies and the main results are presented in §3. In §4, we use simulation experiments to test the performance of a hybrid policy. We show the connection between queueing and knapsack problems in §5. Our conclusion is stated in §6. The technical proofs are collected in the appendix, where we also develop a dynamic programming algorithm to solve the knapsack problem.

2. Multiclass Many-server Queues

We model the system using the fluid model of a $G/M/n+GI$ queueing system with multiple customer classes. Atar et al. (2014) studied a multiclass many-server system under the fixed priority policy. The main difference is that our paper proposes several dynamic priority policies in accordance with more general cost functions. We focus on the analysis of the fluid model and simplify the fluid equations in Atar et al. (2014) benefiting from the assumption of exponential service times.

2.1. A Fluid Model

The model consists of I classes of customers, who arrive at a service system having I unlimited waiting queues and a server pool with a fixed service capacity $n > 0$. For each class $i = 1, \dots, I$, the amount of external arrivals over $[0, t]$ is $E_i(t) = \lambda_i t$, where $\lambda_i > 0$. At time t, the arrival enters the server pool if there is any idle server available. Otherwise, the arrivals who cannot be directly served will join the end of their own queue and are allowed to abandon from the queue once losing patience. We use $Q_i(t)$ and $B_i(t)$ to denote the amount of class i customers waiting in queue and being served in the server pool, respectively. Thus, the total amount of class i customers in the system is $X_i(t) = Q_i(t) + B_i(t)$. Note that in the ED context, the length of each queue cannot be observed in real time since patients normally do not inform hospital staff of their decision to abandon the queue. However, the status of the server pool can be easily observed in real time as the number of patients being treated is surely recorded. For this reason, we will see in §3 that B_i's are important criteria for designing scheduling policies. In the ED setting, customer classes are usually called acuity levels; hereafter, we use these terms interchangeably.

Let $K_i(t)$ denote the total amount of class i customers who have entered service by time t and $D_i(t)$ be the total amount of class i customers who have completed service by time t. It is clear that the cumulative processes $K_i(t)$ and $D_i(t)$ would be nondecreasing. Then, we can deduce the following balance equation for B_i:

$$B_i(t) = B_i(0) + K_i(t) - D_i(t). \tag{1}$$

Let the service time follow the distribution function $G_i(x) = 1 - e^{-\mu_i x}$ for class i customers, namely the service rate of class i customers is μ_i. Due to the memoryless property of exponential distributions, the service completion process satisfies the equation

$$D_i(t) = \mu_i \int_0^t B_i(s) ds. \tag{2}$$

One can see that the derivative of the service completion process is $\mu_i B_i(t)$, which facilitates the analysis of the convergence of the fluid model.

Due to the general patience time distributions, we use the fluid measure-valued process developed in Atar et al. (2014) to capture the dynamics of the queues. Let $\eta_{i,t}([0, x])$ denote the amount of class i customers who have not abandoned by time t with elapsed time since arrival not longer than x no matter whether a customer has entered service or not. Within each queue, customers are served based on the FCFS discipline. Thus the queue length process of class i can be recovered as

$$Q_i(t) = \eta_{i,t}([0, w_i(t)]), \tag{3}$$

where $w_i(t)$ is the waiting time of the customer at the head of the class i queue. Let $R_i(t)$ be the total amount of class i customers who abandon their queue during the time interval $[0,t]$. So we have the following balance equation for Q_i:

$$Q_i(t) = Q_i(0) + E_i(t) - R_i(t) - K_i(t). \tag{4}$$

Let $F_i(\cdot)$ be the patience time distribution of class i customers. Then we have

$$\eta_{i,t}([0,x]) = \int_{t-x}^{t} F_i^c(t-s)dE_i(s), \tag{5}$$

where $F_i^c(\cdot) = 1 - F_i(\cdot)$. Indeed, $dE_i(s)$ is the amount of fluid that enters the system at time s, among which $F_i^c(t-s)dE_i(s)$ is the amount that has not abandoned by time t. For $s < 0$, we regard $dE_i(s)$ as the fluid that had entered the system before time 0. On the other hand, $\eta_{i,t}([0,x])$ only consists of the arrivals between time $t-x$ to t. Thus (5) holds. Clearly, $\eta_{i,t}(dx)$ is the density of class i fluid with the waiting time x but without abandoning at time t. Let the hazard rate function of F_i be $h_i(x) = f_i(x)/F_i^c(x)$. Then $h_i(x)$ is the fraction of the infinitesimal $\eta_{i,t}(dx)$ that abandons the system. Recall that $w_i(t)$ is the longest elapsed time of the fluid in the class i queue at time t, so the total amount of fluid that abandons the system during the interval $[0,t]$ can be written as

$$R_i(t) = \int_0^t \left(\int_0^{w_i(s)} h_i(x)\eta_{i,s}(dx) \right) ds. \tag{6}$$

We denote by Π the class of all work-conserving policies that, for all $t \geq 0$, satisfy

$$\sum_{i=1}^{I} B_i(t) \leq n, \tag{7}$$

$$\big(n - \sum_{i=1}^{I} B_i(t)\big) \sum_{i=1}^{I} Q_i(t) = 0. \tag{8}$$

We refer to equations (1)–(8) as the *fluid model* of a multiclass many-server queueing system. It can be seen from the proof of Theorem 4.3 in Atar et al. (2014) that the tuple (E,B,X,Q,D,K,R,η) satisfying (1)–(8) serves as the fluid limits for many-server systems under any work-conserving policy (see §EC.2.1 for more discussion).

To manage such a system well, the cost it incurs should also be considered. We allow any general nondecreasing function $C_i(\cdot)$ for the queue length cost of each class i. Set $C_i(0) = 0$, which means there won't be any queue length cost once there is no queue. There is also a penalty cost γ_i associated with abandonment for each class i customer. Therefore, for any work-conserving policy $\pi \in \Pi$, the average cost over $[0,T]$ is

$$J_T(\pi) = \frac{1}{T} \sum_{i=1}^{I} \left[\int_0^T C_i\left(Q_i(s)\right) ds + \gamma_i R_i(T) \right]. \tag{9}$$

We define the traffic intensity as $\sum_{i=1}^{I} \lambda_i/\mu_i$. The system is underloaded if $\sum_{i=1}^{I} \lambda_i/\mu_i < n$, critically loaded if $\sum_{i=1}^{I} \lambda_i/\mu_i = n$, or overloaded if $\sum_{i=1}^{I} \lambda_i/\mu_i > n$. Intuitively, if the system is underloaded, then the average cost given above should vanish in the long run under any work-conserving policy. The following theorem validates this intuition.

Theorem 1. *If the system is underloaded, i.e., $\sum_{i=1}^{I} \lambda_i/\mu_i < n$, then for any work-conserving policy $\pi \in \Pi$ the queue length process of each class vanishes after a finite time and the amount of customers being served converges to λ_i/μ_i for each class $i = 1, \dots, I$. As a consequence, the long-run average cost is zero. In other words, there exists a $T > 0$ such that $Q_i(t) = 0$ for all $t > T$,*

$$\lim_{t\to\infty} B_i(t) = \frac{\lambda_i}{\mu_i} \quad \text{and} \quad \lim_{T\to\infty} J_T(\pi) = 0.$$

The proof is postponed to §EC.1. A well-designed scheduling policy is expected to reduce system congestion, especially for an overloaded system. However, a critically loaded system also needs a well-designed scheduling policy. In Mandelbaum and Stolyar (2004), the $Gc\mu$ rule is applied to a queueing system with multiple types of customers and multiskilled servers. Note that their system is critically loaded and the corresponding fluid model is studied under the $Gc\mu$ rule. We go one step further and focus on both the critically loaded and overloaded cases.

The following assumption on the input parameters is required throughout this paper.

Assumption 1 (On Input Parameters). *For each class $i = 1, \dots, I$, the service time distribution $G_i(x) = 1 - e^{-\mu_i x}$ is exponentially distributed and the patience time distribution $F_i(x) = \int_0^x f_i(y)dy$ is strictly increasing. The system is either critically loaded or overloaded, i.e., $\sum_{i=1}^{I} \lambda_i/\mu_i \geq n$. The queue length cost function $C_i(\cdot)$ can be any nondecreasing function and the marginal cost satisfies*

$$\frac{d}{dx} C_i(x) = c_i(x), \tag{10}$$

where $c_i(x) \geq 0$. The abandon penalty cost also satisfies $\gamma_i \geq 0$.

Remark 1. It is well known that the steady-state behavior of the queue length of the fluid model of a single-class many-server queue depends upon the service time distribution only through its mean, but upon the patience time distribution beyond its mean (Whitt (2006)). Therefore, we restrict ourselves to exponential service times. The simulation results in §4 suggest that our proposed policies also works well for nonexponential service times. However, for nonexponential service time distributions, we are not able to prove that the fluid model converges to the invariant state as time goes to infinity. But even for the single-class $G/GI/n+GI$ fluid model, this remains an open problem (see Theorem 2 in Long and Zhang (2014), where an additional assumption on the initial state is needed for critically loaded and overloaded systems).

2.2. Stability and Optimality

We first give the following proposition to show the convergence relationship between the fluid content in the queues and that in service. This would help managers in scheduling the system when the status of the queues or the server pool cannot be fully observed. Usually the situation in waiting rooms in EDs is difficult to observe since the time when patients abandon the queue is normally not observed. This is one of the motivations for designing scheduling policies based on the status of the server pool in §3.

Proposition 1 (Equivalence of the convergence of Q_i and B_i). *Given Assumption 1, for any scheduling policy $\pi \in \Pi$,*

$$Q_i(t) \text{ converges} \Leftrightarrow B_i(t) \text{ converges} \quad \text{for all } i=1,\ldots,I.$$

Moreover, for such a convergent policy, let F_i^{-1} be the inverse function of F_i. Then we have, for all $i=1,\ldots,I$,

$$q_i = \lambda_i \int_0^{F_i^{-1}(1-b_i\mu_i/\lambda_i)} F_i^c(s)ds, \tag{11}$$

where $q_i = \lim\limits_{t\to\infty} Q_i(t)$ and $b_i = \lim\limits_{t\to\infty} B_i(t)$ satisfying $0 \le b_i \le \lambda_i/\mu_i$ and $\sum_{i=1}^I b_i = n$. Therefore, $\lim\limits_{T\to\infty} J_T(\pi) = \sum_{i=1}^I J_i(b_i)$. Here,

$$J_i(b_i) = C_i\big(\lambda_i \int_0^{F_i^{-1}(1-b_i\mu_i/\lambda_i)} F_i^c(s)ds\big) + \gamma_i(\lambda_i - b_i\mu_i). \tag{12}$$

The detailed proof is given in §EC.1.2. The steady-state behavior of customers in the queues and of those being served follows the relation (11), which is consistent with Theorem 3.1 in Whitt (2006). We can see from Proposition 1 that the steady state behavior of the convergence policy has a simple form and the cost function (12) can be expressed in terms of the status of the server pool.

Let us consider the optimization problem in terms of the steady state of the fluid model:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{I} J_i(b_i) \\ \text{subject to} \quad & \sum_{i=1}^{I} b_i \leq n, \\ & 0 \leq b_i \leq \frac{\lambda_i}{\mu_i},\ i = 1, \ldots, I. \end{aligned} \tag{13}$$

The decision variables b_i's can be intuitively understood as the amount of service resources that is assigned to class i customers in the long run. The objective is to minimize the long-run average cost by choosing appropriate b_i's. The first constraint states that b_i's must be chosen so that the amount of customers being served does not exceed the service capacity n. The second constraint implies that at most λ_i/μ_i servers are needed to handle class i customers. Denote by $b^* = (b_1^*, \ldots, b_I^*)$ the optimal solution to this nonlinear programming and J^* the optimal value. It is clear that b^* indicates the optimal allocation of the service capacity. Meanwhile, Proposition 1 implies that J^* is the lower bound of any convergence policies. The main goal of this paper is to find a scheduling policy that attains the lower bound asymptotically.

Definition 1 (Stationary Optimal Control). A control policy $\pi \in \Pi$ is said to be *stationary optimal* if the corresponding cost function (9) satisfies $\lim_{T\to\infty} J_T(\pi) = J^*$.

The following lemma implies that (13) can actually become either a convex or a concave optimization problem.

Lemma 1. *If the queue length cost functions C_i's are convex and the hazard rate functions h_i's are nonincreasing, then the nonlinear programming* (13) *is a convex optimization problem. In contrast, if the queue length cost functions C_i's are concave and the hazard rate functions h_i's are nondecreasing, then the nonlinear programming* (13) *is a concave optimization problem.*

A direct way to show the above lemma is to consider the derivative of the cost function $J_i(b_i)$. By (12) and after some basic calculations, it becomes clear that

$$\frac{d}{db_i}J_i(b_i) = -\frac{c_i\big(\lambda_i \int_0^{F_i^{-1}(1-b_i\mu_i/\lambda_i)} F_i^c(s)ds\big)\mu_i}{h_i(F_i^{-1}(1-b_i\mu_i/\lambda_i))} - \gamma_i\mu_i. \tag{14}$$

We leave the detailed proof to §EC.1. In the following section, we propose different scheduling policies for all types of optimization problems such that the optimal value J^* can be attained in all cases.

3. Scheduling Policies

In this section, we propose dynamic priority scheduling policies that give a time-varying priority order. The goal is to design a policy such that the amount of customers being served approaches b^*. In §3.1 the target-allocation policy is proposed for general queue length cost functions and patience time distributions. We then propose in §3.2 the $Gc\mu/h$ rule, which is an extension to the $Gc\mu$ rule in van Mieghem (1995) by adding abandonments. When the optimization problem (13) is convex, the $Gc\mu/h$ rule is shown to be stationary optimal. On the other hand, if (13) is a concave optimization problem, we find that it is optimal to apply the fixed priority policy in §3.3.

We first introduce the *dynamic priority policy.* At time t, given that there is a certain amount of service resource, the policy chooses some amount of customers from the class with index

$$i \in \underset{i=1,\dots,I}{\arg\max}\, P_i(t), \tag{15}$$

where $P_i(t)$ is the *priority value* for class i at time t and is a continuous function in time t. If the classes of customers with the highest priority value are all in service, then the available service resource can be assigned to classes with the second highest priority value, so on so forth. The stochastic version of (15) is presented in (EC.17). Equivalently, the dynamic priority policy means customers with lower priority can enter service at time t only if at that time no one else in the queue has higher priority. Therefore, the dynamic priority policy can also be expressed as

$$\int_0^t \sum_{\{j=1,\dots,I:P_j(s)>P_i(s)\}} Q_j(s)dK_i(s) = 0, \quad i = 1,\dots,I. \tag{16}$$

Note that $\sum_{\{j=1,\ldots,I:P_j(s)>P_i(s)\}} Q_j(s) = 0$ if $\{j = 1\ldots, I : P_j(s) > P_i(s)\} = \emptyset$. In Lemma EC.2, we prove (16) rigorously. As a special case, the dynamic priority policy becomes the *fixed priority policy* when $P_i(t)$'s are independent of time t. We will see in §3.3 that (16) is actually an extension of (32) in Atar et al. (2014). Moreover, the policy essentially determines the flow rates of customers into the server pool. We present its complete characterization in (EC.20).

3.1. Target-allocation Policy

We propose in this subsection a policy that is suitable for any general queue length cost function and patience time distribution. The optimal solution $b^* = (b_1^*, \ldots, b_I^*)$ of (13) reveals that class i customers should be allocated b_i^* amount of service resources in the long run. Thus we define the following priority value function:

$$P_i(t) = b_i^* - B_i(t) \tag{17}$$

for all $i = 1, \ldots, I$. Intuitively, given the above priority value function, the dynamic priority policy serves the class with largest deviation from its target. Thus more servers will be assigned to those classes of customers who are not given enough service resources. All the B_i's will be gradually close to the optimal allocation b^* of the service capacity. We refer to this control policy as the *target-allocation policy* denoted by π_{b^*}. Its optimality is shown in Theorem 2 below, which is proved in §EC.2.3.

Theorem 2 (Optimality of the Target-allocation Policy). *Given Assumption 1, the fluid model* (1)–(8) *under the target-allocation policy* π_{b^*} *with the priority value function* (17) *satisfies* $\lim_{T\to\infty} J_T(\pi_{b^*}) = J^*$.

3.2. The Generalized $c\mu/h$ Rule

For convex queue length cost functions and patience time distributions with nonincreasing hazard rate functions under which the nonlinear programming (13) becomes a convex optimization by Lemma 1, we propose another dynamic priority policy that is easier to implement. Consider the Lagrangian function

$$L(b_i, \alpha_0, \alpha_i, \beta_i) = \sum_{i=1}^{I} J_i(b_i) - \alpha_0(n - \sum_{i=1}^{I} b_i) - \sum_{i=1}^{I} \alpha_i b_i \mu_i - \sum_{i=1}^{I} \beta_i \cdot (\lambda_i - b_i \mu_i).$$

Combining it with (14), the optimal solution $b^* = (b_1^*, \ldots, b_I^*)$ of (13) solves

$$\begin{aligned}
&\frac{c_i\big(\lambda_i \int_0^{F_i^{-1}(1-b_i^*\mu_i/\lambda_i)} F_i^c(s)ds\big)\mu_i}{h_i(F_i^{-1}(1-b_i^*\mu_i/\lambda_i))} + \gamma_i\mu_i + \alpha_i\mu_i - \beta_i\mu_i = \alpha_0,\\
&\alpha_i b_i^* = 0,\\
&\beta_i \cdot (\lambda_i - b_i^*\mu_i) = 0,\\
&\sum_{i=1}^I b_i^* = n,
\end{aligned}$$

where the Lagrange multipliers satisfy $\alpha_0 \in \mathbb{R}$ and $\alpha_i, \beta_i \geq 0$ for all $i = 1, \ldots, I$. We assume that the cost function C_i, $i = 1\ldots, I$, satisfies conditions that are analogous to (van Mieghem, 1995, Assumption 3) and (Huang et al., 2015, Assumption 2). Specifically, we have the following assumption.

Assumption 2 (Cost Regularity). *The cost function C_i, $i = 1, \ldots, I$, is strictly convex and has an interior solution to the minimization problem* (13)*.*

Recall that the patience time distribution F_i is strictly increasing. By Lemma 1, there is a unique solution to (13) if the cost functions are strictly convex and the hazard rates of patience are nonincreasing. If we assume in addition that $c_i(0) = 0$ and $\gamma_i = 0$, then all customer classes satisfy $b_i^* < \lambda_i/\mu_i$ making $\beta_i = 0$ for all i. Similarly, if we further assume that $h_i(x) \to 0$ as $x \to \infty$, then all customer classes receive positive service resources making $\alpha_i = 0$ for all i. This essentially provides a sufficient condition such that the solution b_i^* is unique and interior.

Under Assumption 2, the Karush-Kuhn-Tucker (KKT) conditions then reduce to

$$\frac{c_i\big(\lambda_i \int_0^{F_i^{-1}(1-b_i^*\mu_i/\lambda_i)} F_i^c(s)ds\big)\mu_i}{h_i(F_i^{-1}(1-b_i^*\mu_i/\lambda_i))} + \gamma_i\mu_i = \alpha_0, \tag{18}$$

$$\sum_{i=1}^I b_i^* = n. \tag{19}$$

Observe that the left hand side of (18) is equal to a constant. This inspires us to consider the following priority value function:

$$P_i(t) = \frac{c_i\big(\lambda_i \int_0^{F_i^{-1}(1-B_i(t)\mu_i/\lambda_i)} F_i^c(s)ds\big)\mu_i}{h_i(F_i^{-1}(1-B_i(t)\mu_i/\lambda_i))} + \gamma_i\mu_i, \tag{20}$$

for all $i = 1, \ldots, I$. This equation is referred to as the priority value function of the *generalized $c\mu/h$ rule* ($Gc\mu/h$) denoted by π_G.

We owe the idea of the $Gc\mu/h$ rule to van Mieghem (1995), where the striking result $Gc\mu$ rule performs well for a single-server multiclass queueing system. In fact, Figure 1 extends the one in van Mieghem (1995) by adding abandonments and considering a many-server pool. Later, the $Gc\mu$ rule was generalized to a system with heterogeneous servers in Mandelbaum and Stolyar (2004). They both consider the conventional diffusion approximation for critically loaded queueing systems without abandonment. We focus on the fluid model of an overloaded multiclass many-server queueing system and allow for customer abandonment. This is why the hazard rate function appears in the priority value function (20). Another main difference is that we take advantage of the equivalence of the convergence of Q_i and B_i (see Proposition 1) to control the system based on the real-time value of $B_i(t)$ instead of $Q_i(t)$. The optimality of our $Gc\mu/h$ rule is shown in the following theorem, which we prove in §EC.2.3.

Theorem 3 (Optimality of the $Gc\mu/h$ rule). *Given Assumptions 1 and 2, if c_i and h_i are differentiable and the hazard rate functions h_i's are nonincreasing, then the fluid model* (1)–(8) *under the $Gc\mu/h$ rule π_G with the priority value function* (20) *satisfies* $\lim_{T\to\infty} J_T(\pi_G) = J^*$.

The assumption that c_i and h_i are differentiable is in the same spirit as the twice differentiability of C_i in §4 of Mandelbaum and Stolyar (2004). It surprised us somewhat that the proofs of the optimality of the target-allocation policy and the $Gc\mu/h$ rule are almost the same. Part of the reason is that the priority value functions go to a constant under both policies—the priority value of the target-allocation policy converges to 0 and that of the $Gc\mu/h$ rule converges to α_0. Therefore, we will prove Theorems 2 and 3 in §EC.2.3 simultaneously.

3.3. Fixed Priority Policy

A fixed priority policy essentially prevents customers from entering service as long as other customers with higher priority are still waiting for their turn. Consider a priority order from class 1 (highest priority) to class I (lowest priority). Then the priority value function in (15) can be specified as

$$P_i(t) = I - i \tag{21}$$

for all $i = 1, \ldots, I$. Note that only if customers with the highest priority value are all in service, then the available service resource can be assigned to classes with the second highest priority value, so on so forth. Equation (16) becomes exactly the same as (32) in Atar et al. (2014). The following proposition shows that the system converges to the steady state under the fixed priority policy (21). Especially, the limit of $B_i(t)$ follows the form as (23), which is the main feature of the fixed priority policy. The proof is postponed to §EC.2.4.

Proposition 2 (Convergence of the Fixed Priority Policy). *Given Assumption 1, the fluid model* (1)–(8) *under the fixed priority policy with the priority value function* (21) *converges to the following steady state*

$$\lim_{t\to\infty} B_i(t) = b_i \quad \text{and} \quad \lim_{t\to\infty} Q_i(t) = q_i \tag{22}$$

starting from any initial state, for all $i = 1, \ldots, I$, where the allocation $b = (b_1, \cdots, b_I)$ of the service capacity to their dedicated classes is

$$b = \Big(\frac{\lambda_1}{\mu_1}, \cdots, \frac{\lambda_{i_0-1}}{\mu_{i_0-1}}, n - \sum_{j<i_0} \frac{\lambda_j}{\mu_j}, 0, \cdots, 0\Big), \tag{23}$$

where $i_0 = \max\Big\{i \in [1, \cdots, n] : \sum_{j=1}^{i-1} \frac{\lambda_j}{\mu_j} < n\Big\}$. And

$$q_i = \begin{cases} 0, & i < i_0, \\ \lambda_i \int_0^{F_i^{-1}(1-b_i\mu_i/\lambda_i)} F_i^c(s)ds, & i = i_0, \\ \lambda_i \int_0^{\infty} F_i^c(s)ds, & i > i_0. \end{cases}$$

Moreover, there exists $T > 0$ such that $Q_i(t) = 0$ for all $t > T$ and $i = 1, \ldots, i_0 - 1$.

The allocation of the service capacity (23) takes a special form such that $b_i = \lambda_i/\mu_i$ for all classes $i < i_0$ being fully served, $b_i = 0$ for all classes $i > i_0$ without receiving any service, and $b_{i_0} = n - \sum_{i=1}^{i_0-1} \lambda_i/\mu_i$ for at most one class i_0 being partially served. This is virtually a solution on the boundary of the feasible region of (13). Therefore, if the nonlinear programming (13) is a concave optimization problem, then the optimal solution $b^* = (b_1^*, \ldots, b_I^*)$ surely has the same form as (23) after reordering the class indices if needed. This is associated with an optimal fixed priority order, of which the corresponding fixed priority policy is denoted by π_{P^*}. Note that the order among the classes with $b_i^* = \frac{\lambda_i}{\mu_i}$ can be arbitrarily determined. It can also be arbitrary for those with $b_i^* = 0$.

Theorem 4 (Optimality of the Fixed Priority Policy). *Given Assumption 1, if the queue length cost functions C_i's are concave and the hazard rate functions h_i's are nondecreasing, then the fluid model* (1)–(8) *under the fixed priority policy π_{P^*} with the priority value function* (21) *(after re-ordering the class indices if needed) satisfies* $\lim_{T\to\infty} J_T(\pi_{P^*}) = J^*$.

Theorem 4 is proved in §EC.2.4. This theorem actually gives a sufficient condition for the optimality of the fixed priority policy. We will show in §5 the innovative connection between the fixed priority policy and knapsack problems.

Remark 2 (Connection to Linear Queue Length Costs and Exponential Patience). We consider a special case of exponential patience time distributions $F_i(x) = 1 - e^{-\theta_i x}$ and linear queue length cost functions by setting $C_i(x) = c_i x$ for all $i = 1, \ldots, I$. Then the optimization problem (13) becomes the following linear programming:

$$\begin{aligned}
\text{minimize} \quad & \sum_{i=1}^{I} \left[c_i \frac{\lambda_i - \mu_i b_i}{\theta_i} + \gamma_i(\lambda_i - \mu_i b_i) \right] \\
\text{subject to} \quad & \sum_{i=1}^{I} b_i \le n, \\
& 0 \le b_i \le \frac{\lambda_i}{\mu_i}, \ i = 1, \ldots, I.
\end{aligned} \tag{24}$$

Let $\tilde{c}_i = c_i + \theta_i \gamma_i$ for notational simplicity. Then the objective function in (24) is identical to

$$\text{maximize} \quad \sum_{i=1}^{I} \frac{\tilde{c}_i \mu_i}{\theta_i} b_i. \tag{25}$$

Due to the simple form of the above objective function, to maximize (25), the obvious solution is to assign as much value (namely λ_i/μ_i) as possible to b_i with higher coefficient $\tilde{c}_i\mu_i/\theta_i$. For convenience, we relabel indices such that $\tilde{c}_1\mu_1/\theta_1 \ge \cdots \ge \tilde{c}_I\mu_I/\theta_I$. After reordering the indices, the linear programming (24) admits an optimal solution with the same form as (23). Thus, it is straightforward to design a fixed priority policy that assigns higher priority to customers with higher $\tilde{c}_i\mu_i/\theta_i$. This is exactly the $c\mu/\theta$ rule studied in Atar et al. (2008, 2010, 2011, 2014). The optimality of the $c\mu/\theta$ rule can be easily seen from Propositions 1 and 2.

4. Numerical Experiments

We first introduce a hybrid policy that is a mixture of the fixed priority policy and the $Gc\mu/h$ rule in §4.1. This policy can be implemented in EDs to reduce the crowding and LWBS rates. We illustrate with performance metrics including the numbers of patients in each of the five acuity levels in steady state and the long-run average cost that the hybrid policy inherits the merits of both the fixed priority policy and the $Gc\mu/h$ rule. In §4.2, we present the parameters used in our experiments. Our simulation results in §4.3 show that the lengths of the queues for levels 1 and 2 patients with the highest priority are close to zero in steady state. We also observe that the patients in the other three less critical levels following the $Gc\mu/h$ rule are able to receive proper medical treatment in the long run.

4.1. A Hybrid Policy

In practice, we can combine the fixed priority policy with the $Gc\mu/h$ rule. It is widely accepted that in EDs patients are generally called for service on a FCFS basis by triage level (Batt and Terwiesch (2015)). However, the fixed priority policy is unfair to the classes of patients with lower priority since they will only be served when all patients with higher priority have been served. The hybrid policy can be used to improve scheduling in EDs such that patients in a lower triage level will also have a chance to be served even when there are still patients in a higher triage level waiting. Meanwhile, the policy always assigns the highest priority to the most critical patients (e.g, levels 1 and 2), ensuring that they receive the quickest response from the physicians. According to the ESI, the hybrid policy can be realized as level 1 with the highest priority, level 2 with the second highest priority and levels 3, 4, and 5 following the $Gc\mu/h$ rule with proper input parameters. The queues of levels 1 and 2 will vanish after a finite time by Proposition 2. This means all patients in levels 1 and 2 will directly enter service and all patients in levels 3, 4, 5 will enter service according to the $Gc\mu/h$ rule. Then by Theorem 3 the fluid model under the hybrid policy converges to a certain steady state.

4.2. Simulation Parameters

In order to demonstrate the fluid approximation, the service capacity is set to be $n = 100$. We now explain the parameters in Table 1. In the column titled "Arrival rate", we display the arrival rates λ_i's for different acuity levels. The service rates μ_i's are set to increase monotonically from level 1 to level 5 as is typically the case in EDs. In general, the monotonicity of the parameters in Table 1 is unnecessary. Since the hybrid policy assigns the highest priority to level 1 and the second highest priority to level 2, there is no need to identify the abandon penalty and queue length cost for these two levels. An alternative way to think about this is that the cost of not treating the most critical patients promptly is high, and so they must be seen by a physician within minutes. We will see in the next subsection that there is almost no queue for levels 1 and 2 patients. For levels 3, 4 and 5 patients, the related costs are presented in the last two columns.

Triage class	Arrival rate	Service rate	Abandon penalty	Queue length cost
	λ_i	μ_i	γ_i	$C_i(x)$
Level 1	30	1	—	—
Level 2	40	2	—	—
Level 3	80	3	3	$3x^2$
Level 4	100	4	2	$2x^2$
Level 5	160	5	1	x^2

Table 1 Arrival and service rates together with related costs for five triage classes

For patients in levels 1 and 2, we assume that they will not abandon the queue because of their high treatment priority. For patients in less critical conditions, their patience time distributions are assumed to be $F_i(x) = 1 - \frac{1}{x+1}$ for all levels $i = 3, 4, 5$, of which the hazard rate function $h_i(x) = \frac{1}{x+1}$ is nonincreasing. Considering the $Gc\mu/h$ rule for levels 3, 4, and 5 and applying the above parameters to (20) yield

$$P_i(t) = 2(6-i)\ln\left(\frac{\lambda_i}{B_i(t)\mu_i}\right)\frac{\lambda_i^2}{B_i(t)} + \gamma_i\mu_i \quad \text{for } i = 3, 4, 5. \tag{26}$$

Thus, once there are no more levels 1 and 2 patients waiting, the patients in levels 3, 4, and 5 will be treated according to the above priority value function.

Assume the arrivals follow Erlang $E_2(1/\lambda_i)$ distributions for levels $i = 1, \ldots, 5$. From now on we use "$E_2(x)$" to denote an Erlang E_2 distribution with mean x, "expo(x)" to denote an exponential distribution with mean x, and "ln(x, y)" to denote a log-normal distribution with mean x and variance y. As pointed out in Remark 1, the steady state of the fluid approximation depends only on the mean of the service time distributions. Thus we simulate the system with three different service time distributions, i.e, expo$(1/\mu_i)$, $E_2(1/\mu_i)$ and $\ln(1/\mu_i, 1/\mu_i^2)$, which have same service rate μ_i for any $i = 1, \ldots, 5$.

With the given parameters and distributions, we run each simulation under the hybrid policy for 1000 time units. The first 10% and the last 10% of the simulation period are regarded as the warm-up and the close-down periods of the system, thus they are discarded when computing the steady state performance metrics. We use the batch-means method with five independent runs to obtain confidence intervals.

4.3. Summary of Results

We present the results of our simulation experiments in this subsection. The steady state of the fluid model under the hybrid policy can be easily computed given the experimental setting in Table 1 and the priority value function (26). For levels 1 and 2 patients with the highest priority, we can deduce from (23) that $b_1 = \frac{\lambda_1}{\mu_1} = 30$ and $b_2 = \frac{\lambda_2}{\mu_2} = 20$. Thus, the service capacity that remains for levels 3, 4 and 5 patients is 50. And their steady state can be obtained by solving the KKT condition (18) with service capacity $b_3 + b_4 + b_5 = 50$. Then the corresponding queue lengths q_i's, $i = 1, \ldots, 5$, and the total cost follow directly from (11) and (12). This yields the fluid approximation of the system, which is displayed in the last column of Table 2 for comparison with the simulation results. In Table 2, we also present the simulation approximations for Q_i's, B_i's and the total long-run average cost along with their relative errors and 95% confidence intervals for three different service time distributions. The relative errors for Q_1 and Q_2 are omitted since their fluid approximations are 0.

It is worth noting that the steady-state performance of the systems with general service times is similar to that of the system with exponential service time distributions. For example, the value of B_3 is 15.758 when service time distributions for different levels are exponential. The corresponding values of B_3 for Erlang E_2 and log-normal distributions are 15.730 and 15.711, respectively. The results of other performance metrics are also close to each other.

Moreover, our approximations using the fluid steady state are fairly accurate. The relative errors of the approximations for Q_i's and B_i's are less than 2.34% and 1.31%, respectively, with an average error of 1.17% for patients who are waiting in queue and 0.59% for

Performance	Exponential expo$(1/\mu_i)$		Erlang $E_2(1/\mu_i)$		Log-normal $\ln(1/\mu_i, 1/\mu_i^2)$		Approx.
	Sim.	Rel.Error(%)	Sim.	Rel.Error(%)	Sim.	Rel.Error(%)	
Q_1	0.600 ±0.063	—	0.555 ±0.076	—	0.578 ±0.130	—	0
Q_2	0.621 ±0.077	—	0.668 ±0.099	—	0.668 ±0.002	—	0
Q_3	42.119 ±1.815	2.34	42.208 ±1.694	2.13	42.325 ±1.643	1.86	43.126
Q_4	49.865 ±1.847	0.91	49.783 ±1.929	1.07	49.816 ±1.904	1.01	50.325
Q_5	80.247 ±3.220	0.48	80.365 ±2.857	0.34	80.497 ±3.233	0.18	80.640
B_1	29.775 ±0.403	0.75	29.864 ±0.500	0.45	29.995 ±0.778	0.02	30
B_2	19.941 ±0.537	0.30	20.024 ±0.181	0.12	20.035 ±0.439	0.18	20
B_3	15.758 ±0.172	1.31	15.730 ±0.060	1.13	15.711 ±0.218	1.01	15.554
B_4	15.245 ±0.171	0.87	15.193 ±0.204	0.52	15.153 ±0.190	0.26	15.114
B_5	19.280 ±0.250	0.27	19.186 ±0.144	0.76	19.145 ±0.218	0.97	19.332
Long run average cost	18027.311 ±562.222	3.66	17833.704 ±414.350	2.55	18050.739 ±556.930	3.80	17390.018

Table 2 Comparison of simulation results and approximations with general service time distributions

patients who are being treated. The quality of the approximations for the long-run average cost is relatively worse. Due to the quadratic queue length cost functions in Table 1, the magnitude of the long-run average cost in the last row of Table 2 is much larger than that of the other performance metrics. Even so, the average error is still less than 3.34% across all simulations with different service time distributions.

5. Knapsack Problems

In this section we show the connection between queueing systems and knapsack problems. The classical 0-1 Knapsack Problem and Fractional Knapsack Problem are reviewed in

§§5.1 and 5.2. We declare that the $c\mu/\theta$ rule derived from (24) is identical to the Fractional Knapsack Problem. In §5.3, we introduce the Fractional 0-1 Knapsack Problem, which turns out to be consistent with the fixed priority scheduling problem in §3.3. To solve it efficiently we propose a dynamic programming algorithm in §EC.3 due to space constraints.

5.1. The 0-1 Knapsack Problem

The 0-1 Knapsack Problem is the most common problem concerning how to pack items into a knapsack without exceeding its capacity to achieve the highest value. Let there be K items, indexed by $k = 1, \ldots, K$, with value v_k and weight w_k for item k. The number of copies of item k will be denoted by x_k being a binary variable equalling 1 if item k is packed in the knapsack and 0 otherwise. The maximum weight that can be carried in the knapsack is W. All values and weights are conventionally assumed to be positive integers. More specifically, we wish to solve the following maximization problem:

$$\begin{aligned} \text{maximize} \quad & \sum_{k=1}^{K} v_k x_k \\ \text{subject to} \quad & \sum_{k=1}^{K} w_k x_k \le W, \\ & x_k \in \{0,1\},\ k = 1, \ldots, K. \end{aligned} \tag{27}$$

The distinctive feature of the 0-1 Knapsack Problem is that the items are indivisible as each x_k is either 0 or 1. The following subsection introduces a continuous version of the problem representing another extreme of the knapsack problem allowing every item to be divided.

5.2. The Fractional Knapsack Problem

Although problem (27) is irrelevant to our queueing model, we state in the following that its continuous version, where the binary constraint $x_k \in \{0,1\}$ is relaxed to $0 \le x_k \le 1$, is equivalent to the $c\mu/\theta$ rule problem (25). Replacing y_k by $w_k x_k$, which represents the weight of item k packed into the knapsack, we can transform the above integer programming to a linear one. Then (27) becomes

$$
\begin{aligned}
\text{maximize} \quad & \sum_{k=1}^{K} \frac{v_k}{w_k} y_k \\
\text{subject to} \quad & \sum_{k=1}^{K} y_k \le W, \\
& 0 \le y_k \le w_k, \ k = 1, \ldots, K.
\end{aligned} \tag{28}
$$

The above problem is the well-known *Fractional Knapsack Problem* named after George Dantzig in Dantzig (1957). Because of its very simple form it admits an immediate algorithm: order the items according to their value-to-weight ratio, $\frac{v_1}{w_1} \ge \cdots \ge \frac{v_K}{w_K}$, then apply a greedy algorithm to pack as many high ratio items into the knapsack as possible. It can be easily seen that the form of the optimal solutions is either 0 or w_k for each item, with at most one exception to choose the fractional part of its weight. Now comparing the maximization problems (25) and (28), there is no doubt that the $c\mu/\theta$ rule is virtually a Fractional Knapsack Problem. We formally state it in the following proposition and omit its proof for brevity.

Proposition 3. *For linear queue length cost functions and exponential patience time distributions, the $c\mu/\theta$ rule problem* (24) *is identical to the Fractional Knapsack Problem* (28)*.*

5.3. The Fractional 0-1 Knapsack Problem

Instead of the linear objective functions in (27) and (28), we consider a nonlinear reward function $V_k(y_k)$ being the reward value of item k with weight y_k packed into the knapsack. For standardization, we set $V_k(0) = 0$. Also $V_k(y_k)$ is postulated to be a nondecreasing function in y_k. Among all the possible choices of $\{y_1, y_2, \cdots, y_K\}$, we allow at most one item to be strictly between 0 and its maximum weight. Hence, the problem (28) is extended to

$$
\begin{aligned}
\text{maximize} \quad & \sum_{k=1}^{K} V_k(y_k) \\
\text{subject to} \quad & \sum_{k=1}^{K} y_k \le W, \\
& 0 \le y_k \le w_k, \ k = 1, \ldots, K, \\
& 0 < y_k < w_k \text{ for at most one } k \in \{1, \cdots, K\}.
\end{aligned} \tag{29}
$$

We refer to (29) as the *Fractional 0-1 Knapsack Problem* since it allows at most one item to be divided like in the Fractional Knapsack Problem and requires other items to be packed in their entirety or not packed at all like in the 0-1 Knapsack Problem. Obviously, the last constraint can be eliminated when (29) is a concave optimization problem. Now it becomes clear that in order to find an optimal fixed priority order it is essential to solve the Fractional 0-1 Knapsack Problem. Therefore, the proposition below immediately follows.

Proposition 4. *For general queue length cost functions and patience time distributions, the fixed priority control problem is equivalent to the Fractional 0-1 Knapsack Problem* (29)*.*

Note that if we restrict ourselves to the family of fixed priority policies then there is no need to require the queue length cost functions to be concave and the hazard rates to be nondecreasing as in Theorem 4. All we need is to find an optimal solution on the boundary of the feasible region of (13) by adding a constraint like the last one in (29).

Remark 3. Note that in the study of knapsack problems, it is quite common to assume that all the weights are integer numbers, i.e., W and w_k in (29) are all integers. It is also well known that the 0-1 Knapsack Problem can be solved in pseudo-polynomial time through dynamic programming (see, e.g., Martello and Toth (1990)). In §EC.3, we develop a dynamic programming algorithm to solve our fixed priority control problem in the same manner, for which we need to assume the related parameters, i.e., λ_i and μ_i in (13), are rational numbers.

6. Conclusion

To the best of our knowledge, this paper is the first to extend the $Gc\mu$ rule by adding abandonment with general patience time distributions. We consider the control problem of a multiclass many-server queueing model with general holding cost functions and patience time distributions based on the fluid approximation. To minimize the queue length costs and abandon penalties, we solve a nonlinear programming in terms of the steady state of the fluid model. The optimal solution inspired us to design three scheduling polices. The target-allocation policy with the priority value function (17) works for any kind of queue length cost functions and patience time distributions. Interestingly, we find that the $Gc\mu/h$ rule with the priority value function (20) is optimal for convex queue length cost functions and nonincreasing hazard rates of patience. In contrast, the fixed priority policy is optimal for concave queue length cost functions and nondecreasing hazard rates of patience with the priority value function (21) (after re-ordering the class indices if needed). In order to find such an optimal order of indices, we develop a dynamic programming algorithm (see §EC.3) based on the unexpected consistency between queueing and knapsack problems. Motivated by the application to EDs, a hybrid of the fixed priority policy and the $Gc\mu/h$ rule is suggested to reduce patient abandonment and crowding in waiting rooms. The simulation results show that the performance of our proposed policy is fairly close to the theoretical result with a relative error of less than 3.8% among all performance metrics.

Several extensions are possible for future research. First, we have assumed that the service time distributions are exponential, which facilitates the equilibrium analysis of the fluid model. The corresponding convergence for the dynamically controlled multiclass many-server queue with nonexponential service time distributions remains to be developed. Another direction is to develop priority value functions based on the waiting time or the queue length. Although we believe that in EDs our proposed dynamic policies based on the number of patients being treated are more realistic, we could accommodate a wider range of situations if we are able to show the asymptotical optimality of a queue length based policy.

Acknowledgments

Nahum Shimkin's research is supported by BSF Grant No. 2006379. Jiheng Zhang's research is supported in part by GRF Grants No. 16201417 and 16501015 from Hong Kong Research Grants Council.

References

Ata, B., I. Gurvich, et al. (2012). On optimality gaps in the halfin-whitt regime. *The Annals of Applied Probability 22*(1), 407–455.

Ata, B. and M. H. Tongarlak (2013). On scheduling a multiclass queue with abandonments under general delay costs. *Queueing Syst.*.

Atar, R. (2005). Scheduling control for queueing systems with many servers: asymptotic optimality in heavy traffic. *Ann. Appl. Probab. 15*(4), 2606–2650.

Atar, R., C. Giat, and N. Shimkin (2008). The $c\mu/\theta$ rule. In *Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools*, ValueTools '08, ICST, Brussels, Belgium, Belgium, pp. 58:1–58:4. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Atar, R., C. Giat, and N. Shimkin (2010). The $c\mu/\theta$ rule for many server queues with abandonment. *Oper. Res. 58*(5), 1427–1439.

Atar, R., C. Giat, and N. Shimkin (2011). On the asymptotic optimality of the $c\mu/\theta$ rule under ergodic cost. *Queueing Syst. 67*(2), 127–144.

Atar, R., H. Kaspi, and N. Shimkin (2014). Fluid limits for many-server systems with reneging under a priority policy. *Math. Oper. Res. 39*(3), 672–696.

Atar, R., A. Mandelbaum, and M. I. Reiman (2004). Scheduling a multi class queue with many exponential servers: asymptotic optimality in heavy traffic. *Ann. Appl. Probab. 14*(3), 1084–1134.

Bassamboo, A. and R. S. Randhawa (2010). On the accuracy of fluid models for capacity sizing in queueing systems with impatient customers. *Oper. Res. 58*(5), 1398–1413.

Bassamboo, A. and R. S. Randhawa (2016). Scheduling homogeneous impatient customers. *Management Science 62*(7), 2129–2147.

Batt, R. J. and C. Terwiesch (2015). Waiting patiently: An empirical study of queue abandonment in an emergency department. *Management Science 61*(1), 39–59.

Burke, G. J., J. Geunes, H. Edwin Romeijn, and A. Vakharia (2008, January). Allocating procurement to capacitated suppliers with concave quantity discounts. *Operations Research Letters 36*(1), 103–109.

Cox, D. and W. Smith (1961). *Queues.* Chapman & Hall/CRC Monographs on Statistics & Applied Probability. Taylor & Francis.

Dai, J. G. and T. Tezcan (2008). Optimal control of parallel server systems with many servers in heavy traffic. *Queueing Syst. 59*(2), 95–134.

Dantzig, G. B. (1957). Discrete-variable extremum problems. *Operations Research 5*(2), pp. 266–277.

Derlet, R. W., R. M. McNamara, A. A. Kazzi, and J. R. Richards (2014). Emergency department crowding and loss of medical licensure: a new risk of patient care in hallways. *Western Journal of Emergency Medicine 15*(2), 137.

Dupuis, P. and R. S. Ellis (1997). *A Weak Convergence Approach to the Theory of Large Deviations.* Wiley Series in Probability and Statistics. Wiley.

Gans, N., G. Koole, and A. Mandelbaum (2003). Telephone call centers: Tutorial, review, and research prospects. *Manufacturing & Service Operations Management 5*(2), 79–141.

Gilboy, N., T. Tanabe, D. Travers, and A. M. Rosenau (2011). Emergency severity index (esi): A triage tool for emergency department. *Rockville, MD: Agency for Healthcare Research and Quality. Retrieved from http://www. ahrq. gov/professionals/systems/hospital/esi/esi1. html.*

Gurvich, I. and W. Whitt (2009a). Queue-and-idleness-ratio controls in many-server service systems. *Math. Oper. Res. 34*(2), 363–396.

Gurvich, I. and W. Whitt (2009b). Scheduling flexible servers with convex delay costs in many-server service systems. *Manufacturing & Service Operations Management 11*(2), 237–253.

Gurvich, I. and W. Whitt (2010). Service-level differentiation in many-server service system via queue-ratio routing. *Oper. Res. 58*(2), 316–328.

Halfin, S. and W. Whitt (1981). Heavy-traffic limits for queues with many exponential servers. *Oper. Res. 29*(3), 567–588.

Harrison, J. M. and M. J. López (1999). Heavy traffic resource pooling in parallel-server systems. *Queueing Syst. 33*(4), 339–368.

Huang, J., B. Carmeli, and A. Mandelbaum (2015). Control of patient flow in emergency departments, or multiclass queues with deadlines and feedback. *Operations Research 63*(4), 892–908.

Kang, W. and K. Ramanan (2010). Fluid limits of many-server queues with reneging. *Ann. Appl. Probab. 20*(6), 2204–2260.

Kaspi, H. and K. Ramanan (2011). Law of large numbers limits for many-server queues. *Ann. Appl. Probab. 21*(1), 33–114.

Kim, J., R. S. Randhawa, and A. R. Ward (2018). Dynamic scheduling in a many-server, multiclass system: The role of customer impatience in large systems. *Manufacturing & Service Operations Management 20*(2), 285–301.

Kim, J. and A. R. Ward (2013). Dynamic scheduling of a $GI/GI/1 + GI$ queue with multiple customer classes. *Queueing Syst.*.

Long, Z. and J. Zhang (2014). Convergence to equilibrium states for fluid models of many-server queues with abandonment. *Oper. Res. Lett. 42*(6–7), 388 – 393.

Long, Z. and J. Zhang (2018). A note on many-server fluid models with time-varying arrivals. *Probability in the Engineering and Informational Sciences, Forthcoming.*

Mandelbaum, A. and A. L. Stolyar (2004). Scheduling flexible servers with convex delay costs: heavy-traffic optimality of the generalized $c\mu$-rule. *Oper. Res. 52*(6), 836–855.

Martello, S. and P. Toth (1990). *Knapsack problems: algorithms and computer implementations.* Wiley-Interscience series in discrete mathematics and optimization. J. Wiley & Sons.

Pines, J. M., J. A. Hilton, E. J. Weber, A. J. Alkemade, H. Al Shabanah, P. D. Anderson, M. Bernhard, A. Bertini, A. Gries, S. Ferrandiz, et al. (2011). International perspectives on emergency department crowding. *Academic Emergency Medicine 18*(12), 1358–1370.

Rowe, B. H., P. Channan, M. Bullard, S. Blitz, L. D. Saunders, R. J. Rosychuk, H. Lari, W. R. Craig, and B. R. Holroyd (2006). Characteristics of patients who leave emergency departments without being seen. *Academic Emergency Medicine 13*(8), 848–852.

Smith, W. E. (1956). Various optimizers for single-stage production. *Naval Research Logistics Quarterly 3*(1-2), 59–66.

van Mieghem, J. A. (1995). Dynamic scheduling with convex delay costs: The generalized $c\mu$ rule. *Ann. Appl. Probab. 5*(3), 809–833.

Whitt, W. (2006). Fluid models for multiserver queues with abandonments. *Oper. Res. 54*(1), 37–54.

Wu, C. A., A. Bassamboo, and O. Perry (2017). Service systems with dependent service and patience times. *Management Science, Forthcoming*.

Zhang, J. (2013). Fluid models of many-server queues with abandonment. *Queueing Syst. 73*(2), 147–193.

Appendix: Dynamic Scheduling for Multiclass Many-server Queues with Abandonment: the Generalized $c\mu/h$ Rule

We prove Theorem 1 in §EC.1.1. The proof of Proposition 1 about the equivalence of the convergence of Q_i and B_i is presented in §EC.1.2. Then we prove Lemma 1 in §EC.1.3. The proofs of the optimality of our proposed scheduling policies is placed in §EC.2. We discuss the prelimit stochastic processes in §EC.2.1. In §EC.2.2, we analyze the flow rates of the fluid model. We provide a proof to the optimality of the target-allocation and the $Gc\mu/h$ rule in §EC.2.3 simultaneously. The optimality of the fixed priority policy is shown in §EC.2.4. In §EC.3, we develop a dynamic programming algorithm to solve the Fractional 0-1 Knapsack Problem.

EC.1. Preliminary Analysis

In this section, we start with the analysis of the fluid model (1)–(8). Due to the fact that class i customers arrive at the system with a constant arrival rate λ_i, we can see from (5) that

$$\eta_{i,t}([0,x]) = \lambda_i \int_0^x F_i^c(s)ds, \tag{EC.1}$$

which implies $\eta_{i,t}(dx) = \lambda_i F^c(x)dx$. This with (6) yields

$$R_i(t) = \lambda_i \int_0^t F_i(w_i(s))ds. \tag{EC.2}$$

For all $i = 1, \ldots, I$, let

$$F_{i,d}(x) := \int_0^x F_i^c(y)dy. \tag{EC.3}$$

Combining (3) and (EC.1) yields $w_i(t) = F_{i,d}^{-1}(Q_i(t)/\lambda_i)$. This together with (EC.2) gives

$$R_i(t) = \lambda_i \int_0^t F_i \left(F_{i,d}^{-1} \left(\frac{Q_i(s)}{\lambda_i} \right) \right) ds. \tag{EC.4}$$

Then it follows from (4) that

$$K_i(t) = \lambda_i \int_0^t F_i^c \left(F_{i,d}^{-1} \left(\frac{Q_i(s)}{\lambda_i} \right) \right) ds - Q_i(t) + Q_i(0). \tag{EC.5}$$

We can also see from (1) and (2) that

$$B_i(t) = B_i(0) + K_i(t) - \mu_i \int_0^t B_i(s) ds,$$

of which the solution can be solved as

$$B_i(t) = B_i(0) e^{-\mu_i t} + \int_0^t e^{-\mu_i (t-s)} dK_i(s).$$

Now plugging (EC.5) into the above equation and applying integration by parts yields

$$X_i(t) = X_i(0) e^{-\mu_i t} + \lambda_i \int_0^t F_i^c \left(F_{i,d}^{-1} \left(\frac{Q_i(t-s)}{\lambda_i} \right) \right) e^{-\mu_i s} ds + \mu_i \int_0^t Q_i(t-s) e^{-\mu_i s} ds. \tag{EC.6}$$

The above equation is consistent with (3.21) in Zhang (2013). It reveals the relationship between Q_i and B_i for each class since $X_i = Q_i + B_i$, and will play a central role in the proofs of Theorem 1 and Proposition 1.

EC.1.1. Underloaded System

If the fluid model is underloaded, i.e., $\sum_{i=1}^I \lambda_i/\mu_i < n$, then any work-conserving policy will be optimal as all the queues vanish in finite time.

Proof of Theorem 1. Let

$$\mathcal{U}(t) = -\sum_{i=1}^I B_i(t) + \sum_{i=1}^I \left[X_i(0) e^{-\mu_i t} + \lambda_i \int_0^t F_i^c \left(F_{i,d}^{-1} \left(\frac{Q_i(t-s)}{\lambda_i} \right) \right) e^{-\mu_i s} ds \right]. \tag{EC.7}$$

Then we can see from (EC.6) that

$$\sum_{i=1}^I Q_i(t) = \mathcal{U}(t) + \sum_{i=1}^I \mu_i \int_0^t Q_i(t-s) e^{-\mu_i s} ds. \tag{EC.8}$$

If $\sum_{i=1}^I Q_i(t)=0$, then by (EC.8), $\mathcal{U}(t)=0-\sum_{i=1}^I \mu_i \int_0^t Q_i(t-s)e^{-\mu_i s}ds \le 0$. If $\sum_{i=1}^I Q_i(t)>0$, then $\sum_{i=1}^I B_i(t)=n$ due to the non-idling constraint (8). Since $\sum_{i=1}^I \lambda_i/\mu_i<n$, we can pick $\delta=(n-\sum_{i=1}^I \lambda_i/\mu_i)/2$, which is positive, such that

$$\begin{aligned}&\sum_{i=1}^I\left[\lambda_i\int_0^t F_i^c\left(F_{i,d}^{-1}\left(\frac{Q_i(t-s)}{\lambda_i}\right)\right)e^{-\mu_i s}ds\right]\\ =&\sum_{i=1}^I\left[\frac{\lambda_i}{\mu_i}\mu_i\int_0^t F_i^c\left(F_{i,d}^{-1}\left(\frac{Q_i(t-s)}{\lambda_i}\right)\right)e^{-\mu_i s}ds\right]\\ \le& n-2\delta,\end{aligned}$$

where the last inequality follows since

$$\mu_i\int_0^t F_i^c\left(F_{i,d}^{-1}\left(\frac{Q_i(t-s)}{\lambda_i}\right)\right)e^{-\mu_i s}ds\le \mu_i\int_0^t e^{-\mu_i s}ds=1-e^{-\mu_i t}\le 1. \tag{EC.9}$$

For this given $\delta>0$, there exists a T_1 such that for all $t>T_1$, $\sum_{i=1}^I X_i(0)e^{-\mu_i t}\le\delta$. Applying theses estimates to (EC.7), we have $\mathcal{U}(t)\le -n+\delta+n-2\delta=-\delta$ for all t satisfying $t>T_1$ and $\sum_{i=1}^I Q_i(t)>0$.

Denote by $\mathcal{S}=\{t\ge 0:\sum_{i=1}^I Q_i(t)>0\}$ the collection of time epochs when the total fluid queue length is larger than 0. Following the discussion of the above two cases, we have that $\mathcal{U}(t)\le 0$ for any $t\in[T_1,+\infty)$ and $\mathcal{U}(t)\le-\delta$ for any $t\in\mathcal{S}\cap[T_1,+\infty)$. We show that $m(\mathcal{S})<\infty$, where m is the Lebesgue measure of real numbers. Consider the contradictory, i.e., $m(\mathcal{S})=\infty$. Note that

$$\begin{aligned}\int_0^\infty e^{-yt}\mathcal{U}(t)dt&=\int_0^{T_1}e^{-yt}\mathcal{U}(t)dt+\int_{T_1}^\infty e^{-yt}\mathcal{U}(t)dt\\ &\le\int_0^{T_1}|\mathcal{U}(t)|dt-\int_{\mathcal{S}\cap[T_1,+\infty)}e^{-yt}\delta dt.\end{aligned}\tag{EC.10}$$

Since we assume $m(\mathcal{S})=\infty$, there exists a $T_2>T_1$ such that $\int_{\mathcal{S}\cap[T_1,T_2]}\delta dt=2+2\int_0^{T_1}|\mathcal{U}(t)|dt$. Choosing $y_0=\frac{\ln 2}{T_2}>0$ yields

$$\int_{\mathcal{S}\cap[T_1,+\infty)}e^{-y_0 t}\delta dt\ge e^{-y_0T_2}\int_{\mathcal{S}\cap[T_1,T_2]}\delta dt=1+\int_0^{T_1}|\mathcal{U}(t)|dt.$$

So we have $\int_0^\infty e^{-y_0 t}\mathcal{U}(t)dt\le -1$ from (EC.10). On the other hand, (EC.8) implies that for all $y>0$,

$$\int_0^\infty e^{-yt}\sum_{i=1}^I Q_i(t)dt=\int_0^\infty e^{-yt}\mathcal{U}(t)dt+\sum_{i=1}^I\left[\int_0^\infty e^{-yt}Q_i(t)dt\cdot\int_0^\infty e^{-yt}\mu_i e^{-\mu_i t}dt\right],$$

where the last term follows from the Laplace transform. Due to the fact that $\int_0^\infty e^{-yt}\mu_i e^{-\mu_i t}dt \le 1$ from (EC.9), the above implies $\int_0^\infty e^{-yt}\mathcal{U}(t)dt \ge 0$ for all $y > 0$, which is a contradiction. Hence, we have shown by contradiction that $m(\mathcal{S}) < \infty$.

Since $m(\mathcal{S}) < \infty$, for any $\varepsilon \in (0,1)$ there exists a $\tau \ge 1$ such that $m(\mathcal{S} \cap [\tau - 1, \infty)) < \varepsilon$. So for any $t \ge \tau$, there exists a $\xi \in [t-\varepsilon, t]$ such that $\sum_{i=1}^I Q_i(\xi) = 0$. The balance equation (4) implies

$$Q_i(t) \le Q_i(\xi) + \lambda_i \varepsilon = \lambda_i \varepsilon \quad \text{for all } t \ge \tau. \tag{EC.11}$$

Denote $X_{i,\tau}(t) := X_i(t+\tau)$ and $Q_{i,\tau}(t) := Q_i(t+\tau)$. In other words, we shift the fluid model by time τ. Similar to (EC.6) we have the following "shifted" version:

$$\begin{aligned}\sum_{i=1}^I X_{i,\tau}(t) &= \sum_{i=1}^I \left[X_i(\tau)e^{-\mu_i t} + \lambda_i \int_0^t F_i^c\left(F_{i,d}^{-1}\left(\frac{Q_{i,\tau}(t-s)}{\lambda_i}\right)\right)e^{-\mu_i s}ds + \mu_i \int_0^t Q_{i,\tau}(t-s)e^{-\mu_i s}ds\right] \\ &\le \sum_{i=1}^I X_i(\tau)e^{-\mu_i t} + \sum_{i=1}^I \frac{\lambda_i}{\mu_i} + \sum_{i=1}^I \lambda_i \varepsilon,\end{aligned}$$

where the inequality is due to (EC.9) and (EC.11). We can see that $X_i(\tau)e^{-\mu_i t} \to 0$ as t goes to infinity. Due to the arbitrariness of ε, taking the limsup on both sides of the above equation yields $\limsup_{t\to\infty} \sum_{i=1}^I X_{i,\tau}(t) = \sum_{i=1}^I \lambda_i/\mu_i < n$. Thus, there must exists a $T > 0$ such that $\sum_{i=1}^I Q_i(t) = 0$ for all $t > T$. Consequently, with regards to (9), we have $\lim_{T\to\infty} J_T(\pi) = 0$ for any work-conserving policy $\pi \in \Pi$. Now by (EC.6), $Q_i(t)$ vanishing in finite time implies the convergence of $B_i(t)$. It can also be seen from (EC.6) that $\lim_{t\to\infty} B_i(t) = \frac{\lambda_i}{\mu_i}$. □

EC.1.2. Equivalence of the convergence of Q_i and B_i

Proposition 1 shows that the convergence of Q_i is equivalent to that of B_i. This helps to control the system based on the status of the server pool especially when the queue length of the system is unobservable. This result will be multiply used in the proofs of the optimality of our scheduling polices.

Proof of Proposition 1. We first prove that the convergence of $Q_i(t)$ implies that of $B_i(t)$. The left-hand side of (EC.6) is nothing but $Q_i(t) + B_i(t)$ and the right-hand side of (EC.6) converges to a certain constant as t goes to infinity due to the convergence of $Q_i(t)$. Therefore, $B_i(t)$ also converges.

Now we start to prove that $B_i(t)$ converging implies the convergence of $Q_i(t)$. Assume that $\lim_{t\to\infty} B_i(t) = b_i$, where the limit b_i must satisfy $b_i \in [0, \lambda_i/\mu_i]$ for all $i \in \mathcal{I}$. Indeed, if $b_i > \lambda_i/\mu_i$, then by (1), (2) and (4),

$$X_i'(t) = \lambda_i - R_i'(t) - \mu_i B_i(t) \le -\frac{1}{2}\mu_i(b_i - \frac{\lambda_i}{\mu_i}) \tag{EC.12}$$

for all large enough t, where $R_i'(t) \ge 0$ following from (6) and the inequality holds due to the assumption $b_i > \lambda_i/\mu_i$. The above implies $X_i(t) \to -\infty$ as t goes to infinity, which is a contradiction. Thus, we have $b_i \le \lambda_i/\mu_i$ for all $i = 1, \ldots, I$. Moreover, there must be $\sum_{i=1}^I b_i = n$. Otherwise, assume to the contrary that $\sum_{i=1}^I b_i < n \le \sum_{i=1}^I \lambda_i/\mu_i$, where the last inequality is due to Assumption 1. This implies there must exist an $i_0 \in \{1, \ldots, I\}$ satisfying $b_{i_0} < \lambda_{i_0}/\mu_{i_0}$. Moreover, there will be $\sum_{i=1}^I B_i(t) < n$ for large enough t, which means all the arrivals enter into service upon arriving. For class i_0, we have for any $\epsilon > 0$ there will be $B_{i_0}(t) \le b_{i_0} + \epsilon$ for all large t. Then by (1) and (2), we have

$$B_{i_0}'(t) = \lambda_{i_0} - \mu_{i_0} B_{i_0}(t) \ge \lambda_{i_0} - \mu_{i_0}(b_{i_0} + \epsilon) \ge \frac{1}{2}(\lambda_{i_0} - \mu_{i_0} b_{i_0})$$

for small enough ϵ. The above implies $B_{i_0}(t) \to +\infty$, which is a contradiction. This proves $\sum_{i=1}^I b_i = n$. Now let

$$X_{i,\infty} := b_i + \lambda_i \int_0^{F_i^{-1}(1-\mu_i b_i/\lambda_i)} F_i^c(s)ds.$$

Plugging (EC.4) into the equation in (EC.12) yields

$$X_i'(t) = \lambda_i F_i^c\left(F_{i,d}^{-1}\left(\frac{X_i(t) - B_i(t)}{\lambda_i}\right)\right) - \mu_i B_i(t).$$

For any $\epsilon > 0$, there exists a $T_0 > 0$ such that for all $t > T_0$, $b_i - \epsilon \le B_i(t) \le b_i + \epsilon$, and as well there exists $\delta_1, \delta_2 > 0$ depending only on ϵ such that

$$X_i'(t) \le -\epsilon \quad \text{whenever } X_i(t) \ge X_{i,\infty} + \delta_1, \tag{EC.13}$$

$$X_i'(t) \ge \epsilon \quad \text{whenever } X_i(t) \le X_{i,\infty} - \delta_2, \tag{EC.14}$$

for all $t \ge T_0$, where δ_1 and δ_2 will be determined in the following. It can be easily checked that

$$\lambda_i F_i^c\left(F_{i,d}^{-1}\left(\frac{X_{i,\infty} - b_i}{\lambda_i}\right)\right) = \mu_i b_i, \tag{EC.15}$$

where $F_{i,d}^{-1}(\cdot)$ is defined in (EC.3). One can find $F_i^c(F_{i,d}^{-1}(\cdot))$ is strictly decreasing. Therefore, when $X_i(t) \geq X_{i,\infty} + \delta_1$, we have

$$X_i'(t) = \lambda_i F_i^c\left(F_{i,d}^{-1}\left(\frac{X_i(t) - B_i(t)}{\lambda_i}\right)\right) - \mu_i B_i(t) \leq \lambda_i F_i^c\left(F_{i,d}^{-1}\left(\frac{X_{i,\infty} - b_i + \delta_1 - \epsilon}{\lambda_i}\right)\right) - \mu_i(b_i - \epsilon).$$

Solving the equation

$$\lambda_i F_i^c\left(F_{i,d}^{-1}\left(\frac{X_{i,\infty} - b_i + \delta_1 - \epsilon}{\lambda_i}\right)\right) - \mu_i(b_i - \epsilon) = -\epsilon$$

yields $\delta_1 = \delta_1(\epsilon) > 0$ following from (EC.15) and the fact that $F_i^c(F_{i,d}^{-1}(\cdot))$ is strictly decreasing. Moreover, $\delta_1(\epsilon) \to 0$ as ϵ goes to zero also following from (EC.15). This determines δ_1 in (EC.13). The δ_2 in (EC.14) can be determined in a same way. Let $\mathcal{L}(t) = (X_i(t) - X_{i,\infty})^2$. Then

$$\mathcal{L}'(t) = 2(X_i(t) - X_{i,\infty})\left[\lambda_i F_i^c\left(F_{i,d}^{-1}\left(\frac{X_i(t) - B_i(t)}{\lambda_i}\right)\right) - \mu_i B_i(t)\right] \leq -2\epsilon \min\{\delta_1, \delta_2\},$$

whenever $X_i(t) \leq X_{i,\infty} - \delta_1$ or $X_i(t) \geq X_{i,\infty} + \delta_2$. So there must be a $T > T_0$ such that $X_i(t) \in (X_{i,\infty} - \delta_1, X_{i,\infty} + \delta_2)$ for all $t > T$. Since δ_1 and δ_2 can be arbitrarily small, we have $\lim_{t\to\infty} X_i(t) = X_{i,\infty}$. Thus $Q_i(t)$ also converges. More specifically, $\lim\limits_{t\to\infty} Q_i(t) = \lambda_i \int_0^{F_i^{-1}(1-\mu_i b_i/\lambda_i)} F_i^c(s)ds$. This implies (11). And we proved that the convergence of B_i and Q_i are equivalent.

In view of (EC.4) and (EC.15), we have $\lim\limits_{T\to\infty} \frac{1}{T} R_i(T) = \mu_i b_i$ for a convergent policy. Thus, the convergence of the total cost $J_T(\pi)$ immediately follows from (9) and satisfies (12) for the cost of each class. □

EC.1.3. Stationary Optimization Problem

In this paper three scheduling polices are proposed to cater the different types of the nonlinear programming (13). Lemma 1 provides sufficient conditions to each type of the optimization problem.

Proof of Lemma 1. It is evident that (14) is a nondecreasing function in b_i for convex cost function C_i and nonincreasing hazard rate function h_i. The reason is simply that $c_i(\lambda_i \int_0^x F_i^c(s)ds)\mu_i/h_i(x)$ is nondecreasing in x, so is the derivative $(d/db_i)J_i(b_i)$. Then the objective function $\sum_{i=1}^I J_i(b_i)$ is a convex function, and the optimization problem (13) is a convex programming.

On the other hand, if the cost function C_i is concave and the hazard rate function h_i is nondecreasing then the objective function $J_i(b_i)$ in (13) is a concave function of b_i. Indeed, it follows that $c_i(\lambda_i \int_0^x F_i^c(s)ds)\mu_i/h_i(x)$ becomes nonincreasing in x. Thus, the derivative $(d/db_i)J_i(b_i)$ is non-increasing in b_i. Therefore the objective function $\sum_{i=1}^I J_i(b_i)$ is a concave function, and the optimization problem (13) becomes a concave optimization.

□

EC.2. Proofs of the Optimality of the Scheduling Policies

EC.2.1. The Prelimit Stochastic Processes

As explained in §2, our fluid model follows directly from Atar et al. (2014). Since they focused on the fixed priority policy, the dynamic priority policy (16) was only proved when the priority value function is specified to be (21). Thus, we still need to prove that (16) holds for any dynamic priority policy.

To this end, let $(E^N, B^N, X^N, Q^N, D^N, K^N, R^N, \eta^N)$ be the prelimit stochastic processes of the fluid limits $(E, B, X, Q, D, K, R, \eta)$ defined in §2. Note that the arrival processes $\{E_i^N : i = 1, \dots, I\}$ are mutually independent renewal processes with mean interarrival times $(\lambda_i^N)^{-1}$, respectively. It's worth pointing out that only η^N the measure-valued process of the buffer is needed since the measure-valued process of the server pool just becomes an auxiliary process due to the exponential service time distribution. The stochastic processes characterize exactly the same dynamics of a multiclass many-server queueing system as that of Atar et al. (2014) except the scheduling policy. In the many-server heavy traffic regime, both the arrival rates λ_i^N, $i = 1, \dots, I$, and the number of agents n^N increase to infinity. More precisely, as $N \to \infty$,

$$\frac{\lambda_i^N}{N} \to \lambda_i, \; i = 1, \dots, I, \quad \text{and} \quad \frac{n^N}{N} \to n.$$

Define the fluid scaled processes $\bar{X}_i^N = N^{-1} X_i^N$ and define $\bar{E}_i^N$, $\bar{B}_i^N$, $\bar{Q}_i^N$, $\bar{D}_i^N$, $\bar{K}_i^N$, $\bar{R}_i^N$ analogously. Similarly, $\bar{\eta}_i^N = N^{-1}\eta_i^N$ for the measure-valued process. Following the same argument as Theorem 4.3 of Atar et al. (2014), we can conclude that for any work-conserving policy there is as $N \to \infty$,

$$(\bar{E}^N, \bar{B}^N, \bar{X}^N, \bar{Q}^N, \bar{D}^N, \bar{K}^N, \bar{R}^N, \bar{\eta}^N) \Longrightarrow (E, B, X, Q, D, K, R, \eta). \tag{EC.16}$$

The following lemma shows that all component functions of the fluid limits are absolutely continuous.

Lemma EC.1. *Consider the fluid model* (1)–(8)*. Then all the fluid processes E_i, B_i, X_i, Q_i, D_i, K_i, R_i, i=1,...,I, are absolutely continuous.*

Proof. It is clear the arrival process E_i is absolutely continuous. The absolute continuity of D_i and R_i follows from (2) and (6), respectively. By (1) and (4), $X_i(t) = X_i(0) + E_i(t) - R_i(t) - D_i(t)$. This implies that X_i is absolutely continuous. As a result, $\sum_{i=1}^I Q_i(t) = (\sum_{i=1}^I X_i(t) - n)^+$ is also absolutely continuous. Then the absolute continuity of $\sum_{i=1}^I K_i(t)$ follows from (4). Since the entrance into service process $K_i(t)$ is nondecreasing, it follows that each K_i must be absolutely continuous. Consequently, the absolute continuity of B_i and Q_i follows from (1) and (4). This completes the proof. □

In the Nth system, let $P_i^N(t)$ be the priority value function of each level. Then the stochastic version of the fluid *dynamic priority policy* (15) is said to be: at time t, given that a customer is to be served by an idle server, it chooses the head-of-the-line customer from the class with index

$$i \in \arg\max_{i=1,\ldots,I} P_i^N(t), \tag{EC.17}$$

where $P_i^N(t)$ is the *priority value* for class i at time t of the Nth system. If queue i with the highest priority value is empty, the idle server will check classes with the second largest priority value, so on so forth. Ties are broken arbitrarily once there are multiple queues with same priority value, for example, in favor of the smallest index i. It can be easily seen that the stochastic dynamic priority policy (EC.17) is equivalent to

$$\int_0^t \sum_{\{j=1,\ldots,I:P_j^N(s)>P_i^N(s)\}} Q_j^N(s)dK_i^N(s) = 0, \quad i = 1,\ldots,I. \tag{EC.18}$$

Note that $\sum_{\{j=1,\ldots,I:P_j^N(s)>P_i^N(s)\}} Q_j^N(s) = 0$ if $\{j = 1,\ldots,I : P_j^N(s) > P_i^N(s)\} = \emptyset$.

We end up this subsection by proving (16) based on the above stochastic dynamic priority policy.

Lemma EC.2. *If $P_i^N(t) \Rightarrow P_i(t)$, i=1,...,I, as N goes to infinity for some continuous priority value function $P_i(t)$, then* (16) *holds.*

Proof. By Lemma EC.1, let $K_i'(t) = (d/dt)K_i(t)$. It suffices to prove that $K_i'(t) = 0$ if $\sum_{\{j:P_j(t)>P_i(t)\}} Q_j(t) > 0$, which gives (16). So assume that there exists $t > 0$ and $j \in \{1,\ldots,I\}$ such that $P_j(t) > P_i(t)$ and $Q_j(t) > 0$. Due to the continuity of P_j and P_i from the condition of this lemma and the continuity of Q_j by Lemma EC.1, we can conclude that for N large enough $P_j^N(s) > P_i^N(s)$ and $\bar{Q}_j^N(s) > 0$ for $|s-t| < \delta$ and some $\delta > 0$. According to the stochastic dynamic priority policy (EC.17) (or equivalently (EC.18)), $\bar{K}_i^N(t+\delta) - \bar{K}_i^N(t-\delta) = 0$, and therefore $K_i(t+\delta) - K_i(t-\delta) = 0$ following from (EC.16). This gives the desired result. □

EC.2.2. Flow Rates of the Fluid Model

The following lemma extends Theorem 3.2 in Atar et al. (2014) and characterizes a notable property of the dynamic priority policy that the entrance into service process can be represented by the external arrival and departure processes.

Let ${}^*I_j(t)$ be the collection of indices with the first jth highest priority value at time t recursively defined as follows:

$$ {}^*I_1(t) = \arg\max_{i\in\{1,\dots,I\}} P_i(x), \tag{EC.19}$$

and for $1 \le j \le I$,

$$ {}^*I_{j+1}(t) = {}^*I_j(t) \cup \arg\max_{i\in\{1,\dots,I\}\backslash {}^*I_j(t)} P_i(t).$$

Lemma EC.3. *Consider the fluid model* (1)–(8) *given any continuous priority value function $P_i(t)$. Then the entrance into service processes $K_i(t)$ are absolutely continuous, and the derivatives $K_i'(t) := (d/dt)K_i(t)$ satisfy a.e. for $j = 1, \dots, I$,*

$$\sum_{i\in {}^*I_j(t)} K_i'(t) = \begin{cases} \sum_{i=1}^I \mu_i B_i(t) & \text{if } \sum_{i\in {}^*I_j(t)} Q_i(t) > 0, \\ [\sum_{i=1}^I \mu_i B_i(t)] \wedge \sum_{i\in {}^*I_j(t)} \lambda_i & \text{if } \sum_{i\in {}^*I_j(t)} Q_i(t) = 0, \sum_{i=1}^I B_i(t) = n, \\ \sum_{i\in {}^*I_j(t)} \lambda_i & \text{if } \sum_{i=1}^I B_i(t) < n, \end{cases} \tag{EC.20}$$

where $a \wedge b$ is the minimum of a and b.

Proof. We prove this lemma following a similar argument to Theorem 3.2 in Atar et al. (2014). The absolutely continuity of K_i has been proven in Lemma EC.1.

If $\sum_{i=1}^I B_i(t) < n$ for some t, then by the continuity of B_i's (which follows from (1) using the continuity of K_i and D_i) this holds on a neighborhood of t. For any s in such a neighborhood, it is easily seen that $Q_i(s) = 0$ by (8) and $R_i'(s) = 0$ by (EC.4). Hence, by (4), we have $K_i(s) - K_i(t) = E_i(s) - E_i(t)$. This shows $K_i'(t) = \lambda_i$ for all $i = 1, \dots, I$.

On the other hand, if $\sum_{i\in {}^*I_j(t)} Q_i(t)>0$, then we have $\sum_{i\in {}^*I_j(t)} Q_i(s)>0$ for any $s\geq t$ in a right neighborhood of t by the continuity of Q_i's (which follows from (4) using the continuity of E_i, R_i, and K_i). By (8), for any s in such a neighborhood, $\sum_{i=1}^I B_i(s)=n$. We also have ${}^*I_j(s)\subset {}^*I_j(t)$ for small enough neighborhood, which is due to continuity of the priority value function. According to the definition of the dynamic priority policy (15), customers with lower priority value can be served only if those with higher priority are all in service. This together with the fact $\sum_{i\in {}^*I_j(t)} Q_i(s)>0$ implies that there must be $K_i'(s)=0$ for all $i\notin {}^*I_j(t)$ for small enough neighborhood. It then follows from (1) that

$$\sum_{i\in {}^*I_j(t)} K_i(s)-\sum_{i\in {}^*I_j(t)} K_i(t)=\sum_{i=1}^I D_i(s)-\sum_{i=1}^I D_i(t).$$

By (2), the above implies that $\sum_{i\in {}^*I_j(t)} K_i'(t)=\sum_{i=1}^I \mu_i B_i(t)$ if $\sum_{i\in {}^*I_j(t)} Q_i(t)>0$.

Now we start to prove the second entry in (EC.20). Since $\sum_{i=1}^I B_i(t)$ and $\sum_{i\in {}^*I(t)} Q_i(t)$ are absolutely continuous, it follows that $\sum_{i=1}^I B_i'(t)=0$ a.e. on $S_1:=\{t:\sum_{i=1}^I B_i(t)=n\}$ and $\sum_{i\in {}^*I(t)} Q_i'(t)=0$ a.e. on $S_2:=\{t:\sum_{i\in {}^*I(t)} Q_i(t)=0\}$ by Theorem A.6.3 in Dupuis and Ellis (1997). Moreover, from (1) and (4) we have

$$\sum_{i=1}^I B_i'(t)=\sum_{i=1}^I K_i'(t)-\sum_{i=1}^I \mu_i B_i(t),$$
$$\sum_{i\in {}^*I_j(t)} Q_i'(t)=\sum_{i\in {}^*I_j(t)} \lambda_i-\sum_{i\in {}^*I_j(t)} K_i'(t)-\sum_{i\in {}^*I_j(t)} R_i'(t).$$

Note that $R_i'(t)=0$ whenever $Q_i(t)=0$ by (EC.4). Thus a.e. on $S_1\cap S_2$, we have $\sum_{i=1}^I K_i'(t)=\sum_{i=1}^I \mu_i B_i(t)$ and $\sum_{i\in {}^*I_j(t)} K_i'(t)=\sum_{i\in {}^*I_j(t)} \lambda_i$. Hence a.e. on $S_1\cap S_2$, $\sum_{i\in {}^*I_j(t)} K_i'(t)=\sum_{i\in {}^*I_j(t)} \lambda_i(t)=[\sum_{i=1}^I \mu_i B_i(t)]\wedge\sum_{i\in {}^*I_j(t)} \lambda_i$. This completes the proof. □

EC.2.3. Optimality of the Target-allocation Policy and the $Gc\mu/h$ Rule

In view of the fact that the priority value functions go to an equal constant under both policies. We will see that the proofs of the optimality of the target-allocation policy and the $Gc\mu/h$ rule are exactly the same. Thus we prove Theorems 2 and 3 simultaneously, which is presented in the end of this subsection. Before that, some auxiliary Lemmas EC.4 – EC.7 are analyzed. First we introduce the following auxiliary functions.

For the target-allocation policy π_{b^*} proposed in §3.1, let

$$A_i(x)=\alpha_0+b_i^*-x, \tag{EC.21}$$

where α_0 can be chosen as any constant. In order to have a same proof as the optimality of the $Gc\mu/h$ rule, we choose α_0 to be the one in (18). With a little bit abuse of notation, for the $Gc\mu/h$ rule, we also introduce $A_i(\cdot)$ as follows:

$$A_i(x) = \frac{c_i\big(\lambda_i \int_0^{F_i^{-1}(1-x\mu_i/\lambda_i)} F_i^c(u)du\big)\mu_i}{h_i(F_i^{-1}(1-x\mu_i/\lambda_i))} + \gamma_i\mu_i. \tag{EC.22}$$

Note that by (18) and (EC.21), we have

$$A_i(b_i^*) = \alpha_0 \tag{EC.23}$$

for both $A_i(\cdot)$ in (EC.21) and (EC.22). Obviously, $A_i(\cdot)$ in (EC.21) is strictly decreasing. And $A_i(\cdot)$ in (EC.22) is also a strictly decreasing function under Assumption 2. Thus, within this subsection $A_i(x)$ could be either (EC.21) or (EC.22). Now introduce

$$^*A(B(t)) := \max_{i=1,\ldots,I} A_i(B_i(t)). \tag{EC.24}$$

In view of (17) and (EC.21), for the target-allocation policy, we can consider $A_i(B_i(t))$ as the priority value function instead of the one in (17). Then $^*I_1(t)$ in (EC.19) can be replaced by

$$^*I_1(t) := \{i \in \{1,\ldots,I\} : A_i(B_i(t)) = {}^*A(B(t))\}, \tag{EC.25}$$

which is the collection of indices with the highest priority value at time t. And define

$$^*B_i(t) \doteq \{\zeta \geq 0 : A_i(\zeta) = {}^*A(B(t))\}. \tag{EC.26}$$

Lemma EC.4. *Consider the fluid model* (1)–(8) *given the priority value function* (17) *or* (20)*. The following properties hold at any time $t \geq 0$.*

(1) *The process $B_i(t)$ is absolutely continuous and the derivative $B_i'(t) := (d/dt)B_i(t)$ satisfies a.e.*

$$\sum_{i\in {}^*I_1(t)} B_i'(t) \geq 0. \tag{EC.27}$$

(2) *Moreover, if $\sum_{i=1}^I {}^*B_i(t) \leq n - \delta$, for some $\delta > 0$, then there exists a constant $\epsilon_0 > 0$ depending only on δ such that*

$$B_i(t) \leq b_i^* - \epsilon_0 \quad \textit{for all } i \in {}^*I(t), \tag{EC.28}$$

and there also exists a constant $\epsilon_1 > 0$ depending only on δ such that

$$\sum_{i\in {}^*I_1(t)} B_i'(t) \geq \epsilon_1. \tag{EC.29}$$

Proof. First, the absolute continuity of $B_i(t)$ follows from (1) and Lemma EC.3. Now, we claim that there must be $B_i(t) \le b_i^*$ for all $i \in {}^*I(t)$. Suppose there exists an $i_0 \in {}^*I(t)$ satisfying $B_{i_0}(t) > b_{i_0}^*$. Together this with (EC.23) yields $A_{i_0}(B_{i_0}(t)) \le A_{i_0}(b_{i_0}^*) = \alpha_0$. By (EC.25), this implies ${}^*A(B(t)) \le \alpha_0$, which yields $A_i(B_i(t)) \le \alpha_0$ for all $i \in \{1,\dots,I\}$. Thus $B_i(t) \ge b_i^*$ for all $i \in \{1,\dots,I\}$ following from (EC.23). Due to the strict inequality of $B_{i_0}(t) > b_{i_0}^*$, we obtain $\sum_{i=1}^I B_i(t) > n$. This contradicts (7) and then it follows $B_i(t) \le b_i^*$ for all $i \in {}^*I_1(t)$. From (1),

$$\sum_{i\in {}^*I_1(t)} B_i'(t) = \sum_{i\in {}^*I_1(t)} K_i'(t) - \sum_{i\in {}^*I_1(t)} D_i'(t). \tag{EC.30}$$

By Lemma EC.3, the above expression is nonnegative once $\sum_{i\in {}^*I_1(t)} K_i'(t) = \sum_{i=1}^I D_i'(t) = \sum_{i=1}^I \mu_i B_i(t)$. So we just need to consider the other possible case $\sum_{i\in {}^*I_1(t)} K_i'(t) = \sum_{i\in {}^*I_1(t)} \lambda_i$ when proving (EC.27), which still holds since $D_i'(t) = B_i(t)\mu_i \le b_i^*\mu_i \le \lambda_i$ for all $i \in {}^*I_1(t)$. Thus (EC.27) holds.

We show that the condition $\sum_{i=1}^I {}^*B_i(t) \le n - \delta$ implies there exists an $\epsilon' > 0$, such that

$$B_i(t) \le b_i^* - \epsilon' \quad \text{for all } i \in {}^*I_1(t), \tag{EC.31}$$

where ϵ' depends only on the subset ${}^*I_1(t)$ and δ. Indeed, there must be $B_i(t) < b_i^*$ for all $i \in {}^*I_1(t)$ with strict inequalities. Otherwise, we will have $B_i(t) = b_i^*$ for at least one $i \in {}^*I_1(t)$, which causes ${}^*A(B(t)) = \alpha_0$ following from (EC.23) and (EC.25). Then ${}^*B_i(t) = b_i^*$ for all $i \in \mathcal{I}$ deducing from (EC.26). This is a contradiction to the assumption $\sum_{i=1}^I {}^*B_i(t) < n$ since $\sum_{i=1}^I b_i^* = n$. Therefore ${}^*A(B(t)) = \alpha_0 + \varepsilon$, for some $\varepsilon > 0$. From (EC.26), we have

$$\sum_{i=1}^I {}^*B_i(t) = \sum_{i=1}^I A_i^{-1}(\alpha_0 + \varepsilon) \le s - \delta.$$

Let ε^* satisfy $\sum_{i=1}^I A_i^{-1}(\alpha_0 + \varepsilon^*) = n - \delta$. There must be $0 < \varepsilon^* \le \varepsilon$ since A_i^{-1}, $i \in \{1,\dots,I\}$, are decreasing. By (EC.25), for all $i \in {}^*I_1(t)$, $B_i(t) = A_i^{-1}(\alpha_0 + \varepsilon) \le A_i^{-1}(\alpha_0 + \varepsilon^*) = b_i^* - (b_i^* - A_i^{-1}(\alpha_0 + \varepsilon^*))$. Now let $\epsilon' = \min_{i\in {}^*I_1(t)}(b_i^* - A_i^{-1}(\alpha_0 + \varepsilon^*))$ which is positive and depends only on the subset ${}^*I_1(t) \subset \{1,\dots,I\}$ and δ. This proves (EC.31). Because there is only a finite number of subsets of $\{1,\dots,I\}$, we have proved (EC.28) and ϵ_0 only depends on δ.

From (2) and (EC.30), if $\sum_{i\in {}^*I_1(t)} K_i'(t) = \sum_{i\in {}^*I_1(t)} \lambda_i$, then

$$\sum_{i\in {}^*I_1(t)} B_i'(t) = \sum_{i\in {}^*I_1(t)} \lambda_i - \sum_{i\in {}^*I_1(t)} B_i(t)\mu_i$$

$$
\begin{aligned}
&\geq \sum_{i\in {}^*I_1(t)} \lambda_i - \sum_{i\in {}^*I_1(t)} (b_i^* - \epsilon_0)\mu_i \\
&\geq \sum_{i\in {}^*I_1(t)} \mu_i \epsilon_0 \\
&\geq \min_{i\in\{1,\dots,I\}} \mu_i \epsilon_0,
\end{aligned}
$$

where the first inequality uses (EC.28), the second inequality is due to the fact $\lambda_i \geq b_i^* \mu_i$. Another case is $\sum_{i\in {}^*I_1(t)} K_i'(t) = \sum_{i=1}^I D_i'(t)$, which happens only when $\sum_{i=1}^I B_i(t) = n$ deduced from Lemma EC.3. In this case the set $\{1,\dots,I\}\setminus {}^*I(t)$ is nonempty, otherwise, observing (EC.28), $\sum_{i=1}^I B_i(t) = \sum_{i\in {}^*I_1(t)} B_i(t) < \sum_{i\in {}^*I_1(t)} b_i^* \leq n$ becoming a contradiction. Then there must be an $i_1 \in \{1,\dots,I\}\setminus {}^*I_1(t)$ satisfying $B_{i_1}(t) \geq b_{i_1}^*$. Thus, by (EC.30),

$$
\sum_{i\in {}^*I_1(t)} B_i'(t) = \sum_{i\in\{1,\dots,I\}\setminus {}^*I_1(t)} D_i'(t) \geq B_{i_1}(t)\mu_{i_1} \geq b_{i_1}^*\mu_{i_1} \geq \min_{i\in\{1,\dots,I\}} b_i^*\mu_i.
$$

Combining the above two inequalities yields (EC.29). □

It follows from (EC.21) and the absolutely continuous of $B_i(t)$ proved in Lemma EC.4 that $A_i(B_i(t))$ is absolutely continuous for the target-allocation policy. For the $Gc\mu/h$ rule, with the fact c_i and h_i are differentiable assumed in Theorem 3, the function $A_i(x)$ in (EC.22) is absolutely continuous. Thus $A_i(B_i(t))$ is also absolutely continuous for the $Gc\mu/h$ rule. This implies that ${}^*A(B(t))$ is absolutely continuous, so is ${}^*B_i(t)$ by (EC.26). Let us call such points t *strictly regular*. This concept was also used in Mandelbaum and Stolyar (2004) (see Page 847 for reference).

Lemma EC.5. *Consider the fluid model* (1)–(8) *given the priority value function* (17) *or* (20). *Suppose t is a strictly regular point, then*

$$
\frac{d}{dt}[A_i(B_i(t))] = \frac{d}{dt}[{}^*A(B(t))] \quad \textit{for all } i\in {}^*I_1(t). \tag{EC.32}
$$

Proof. Suppose contrarily

$$
\frac{d}{dt}[A_{i_0}(B_{i_0}(t))] = \max_{i\in {}^*I_1(t)} \frac{d}{dt}[A_i(B_i(t))] > \min_{i\in {}^*I_1(t)} \frac{d}{dt}[A_i(B_i(t))] = \frac{d}{dt}[A_{i_1}(B_{i_1}(t))]
$$

for some $i_0, i_1 \in {}^*I_1(t)$. There exist sequences $\{\epsilon_1^n, \epsilon_2^n\}$ both converging to 0 such that $A_{i_0}(B_{i_0}(t+\epsilon_1^n)) > A_{i_1}(B_{i_1}(t+\epsilon_1^n))$ and $A_{i_0}(B_{i_0}(t-\epsilon_2^n)) < A_{i_1}(B_{i_1}(t-\epsilon_2^n))$. Thus $\lim\limits_{s\to t+} \frac{{}^*A(B(s)) - {}^*A(B(t))}{s-t} = \lim\limits_{\epsilon_1^n\to 0} \frac{A_{i_0}(B_{i_0}(t+\epsilon_1^n)) - A_{i_0}(B_{i_0}(t))}{\epsilon_1^n} = \frac{d}{dt}[A_{i_0}(B_{i_0}(t))]$. Similarly, $\lim\limits_{s\to t-} \frac{{}^*A(B(s)) - {}^*A(B(t))}{s-t} = \frac{d}{dt}[A_{i_1}(B_{i_1}(t))] \neq \frac{d}{dt}[A_{i_0}(B_{i_0}(t))]$, which contradicts the strict regularity at t. This completes the proof. □

Lemma EC.6. *Consider the fluid model* (1)–(8) *given the priority value function* (17) *or* (20)*. The following inequalities hold for almost all $t \geq 0$,*

$$^*A(B(t)) \geq \alpha_0, \tag{EC.33}$$

$$\frac{d}{dt}[^*A(B(t))] \leq 0. \tag{EC.34}$$

And if $\sum_{i=1}^I {}^*B_i(t) \leq n - \delta$, for some $\delta > 0$, then there exists an $\epsilon_1 > 0$ depending only on δ such that for almost all $t > 0$,

$$\frac{d}{dt}[\sum_{i=1}^I {}^*B_i(t)] \geq \epsilon_1, \tag{EC.35}$$

where ϵ_1 is given in (EC.29)*.*

Proof. In view of (7) and the fact that $\sum_{i=1}^I b_i^* = n$, there must be an $i \in \{1, \ldots, I\}$ such that $B_i(t) \leq b_i^*$. Then by (EC.23) and (EC.24) the inequality (EC.33) follows.

We have shown in the above of Lemma EC.5 that $^*A(B(t))$ and $^*B_i(t)$ for all $i = 1, \ldots, I$ are absolutely continuous, which means they have derivatives almost everywhere. Consider an arbitrary strictly regular point $t > 0$. We cannot have $(d/dt)^*A(B(t)) > 0$ since by Lemma EC.5 this would imply $B_i'(t) < 0$ for all $i \in {}^*I_1(t)$. This contradicts (EC.27). So we have (EC.34).

Next we prove (EC.35). Using (EC.25), (EC.26), and (EC.32) yields $^*B_i'(t) = B_i'(t)$ for all $i \in {}^*I_1(t)$. Therefore,

$$\sum_{i=1}^I {}^*B_i'(t) \geq \sum_{i \in {}^*I_1(t)} {}^*B_i'(t) = \sum_{i \in {}^*I_1(t)} B_i'(t) \geq \epsilon_1,$$

where the first inequality comes from the fact that $^*B_i'(t) \geq 0$ for all $i = 1, \ldots, I$ (which is implied by (EC.34)) and the second inequality follows from (EC.29). □

The following lemma is similar to Proposition 7 in van Mieghem (1995), which is essentially a sufficient condition of the optimality of our policies.

Lemma EC.7. *Consider the fluid model* (1)–(8) *given the priority value function* (17) *or* (20)*. If*

$$\max_{1 \leq k,l \leq I} |A_k(B_k(t)) - A_l(B_l(t))| \to 0 \quad \textit{as } t \to \infty, \tag{EC.36}$$

then the amount of customers in service $B_i(t)$ satisfies $\lim_{t \to \infty} B_i(t) = b_i^*$ for all $i = 1, \ldots, I$.

Proof. We first claim that for any $\epsilon_0 > 0$ and $i \in \{1, \dots I\}$,

$$B_i(t) \le \lambda_i/\mu_i + \epsilon_0 \quad \text{for large enough } t. \tag{EC.37}$$

Otherwise, there must be an $i_0 \in \{1, \dots, I\}$ and a subsequence $t_n \to \infty$ such that

$$B_{i_0}(t_n) > \lambda_{i_0}/\mu_{i_0} + \epsilon_0 \ge b^*_{i_0} + \epsilon_0. \tag{EC.38}$$

We have $A_{i_0}(B_{i_0}(t_n)) \le A_{i_0}(b^*_{i_0}) = \alpha_0$ by (EC.23) and the fact that $A_{i_0}(\cdot)$ is decreasing. Then by (EC.36), $A_i(B_i(t_n)) \le \alpha_0 + \epsilon'$ for all $i \ne i_0$ and large enough t_n, where $\epsilon' > 0$ could be arbitrarily small. Thus we can chose ϵ' small enough such that for all $i \ne i_0$, $B_i(t_n) \ge b^*_i - \epsilon_0/(2(I-1))$ for large enough t_n. This together with the assumption (EC.38) yields $\sum_{i=1}^I B_i(t_n) \ge \sum_{i=1}^I b^*_i + \epsilon_0/2 > n$, contradicting (7). Thus (EC.37) holds.

Now we use (EC.37) to prove

$$\lim_{t\to\infty} \sum_{i=1}^I B_i(t) = n. \tag{EC.39}$$

To this end, we show that for any $\varepsilon > 0$ there exists a $\delta > 0$ such that

$$\sum_{i=1}^I B'_i(t) \ge \delta \quad \text{whenever } \sum_{i=1}^I B_i(t) \le n - \varepsilon. \tag{EC.40}$$

Since $\sum_{i=1}^I \lambda_i/\mu_i \ge n$, there must exist $i_1 \in \{1, \dots, I\}$ such that $B_{i_1}(t) \le \frac{\lambda_{i_1}}{\mu_{i_1}} - \frac{\varepsilon}{2I}$. Then we can choose the ϵ_0 in (EC.37) small enough such that

$$\sum_{i=1}^I D'_i(t) = \sum_{i \ne i_1} \mu_i B_i(t) + \mu_{i_1} B_{i_1}(t) \le \sum_{i=1}^I \lambda_i - c\varepsilon,$$

where c is a small enough constant. Note that $\sum_{i=1}^I K'_i(t) = \sum_{i=1}^I \lambda_i$ whenever $\sum_{i=1}^I B_i(t) < n$ by (EC.20). Thereby, $\sum_{i=1}^I B'_i(t) \ge c\varepsilon$ is strictly positive deduced from the above and (1). Let $\delta = c\varepsilon$, then (EC.40) holds. This yields (EC.39).

Next we consider the following two cases:

Case 1: $A_i(x)$ is given in (EC.21). Fix a class, say $l \in \{1, \dots, I\}$. Then by (EC.21) and (EC.36),

$$\lim_{t\to\infty} |b^*_k - B_k(t) - (b^*_l - B_l(t))| = 0.$$

Summing over the classes $k=1,\dots,I$,

$$\lim_{t\to\infty}\Big|\sum_{k=1}^{I}(b_k^*-B_k(t))-I\cdot(b_l^*-B_l(t))\Big|=0.$$

From (EC.39), the above implies $B_l(t)\to b_l^*$. Thus, $B_i(t)\to b_i^*$ for all $i=1,\dots,I$.

Case 2: $A_i(x)$ is given in (EC.22). We also fix a class, say $l\in\{1,\dots,I\}$. The limit (EC.36) shows that for all $\epsilon_1>0$ there exists a T such that for all $t>T$,

$$|A_k(B_k(t))-A_l(B_l(t))|<\epsilon_1\quad\text{for all } k\in\{1,\dots,I\}.$$

Since $A_k(x)$ is strictly decreasing and continuous in x according to (EC.22), its inverse A_k^{-1} is also strictly decreasing and continuous. Thus by (EC.37) and the above, for all $\epsilon>0$ there exists a $\delta'>0$ such that if $\epsilon_0,\epsilon_1<\delta'$, then

$$\left|B_k(t)-A_k^{-1}(A_l(B_l(t)))\right|<\epsilon.$$

Summing over the classes $k=1,\dots,I$,

$$\left|\sum_{k=1}^{I}B_k(t)-\sum_{k=1}^{I}A_k^{-1}(A_l(B_l(t)))\right|<\epsilon I.$$

Because the function $\sum_{k=1}^{I}A_k^{-1}(A_l(\cdot))$ is strictly decreasing , $B_l(t)$ converges by (EC.39). The policy satisfying (EC.36) controls the service capacity such that $b^*=(b_1^*,\dots,b_I^*)$ is the solution to the sufficient first order conditions of the minimization problem (13). Thus, $B_i(t)\to b_i^*$ for all $i=1,\dots,I$. Combining the above two cases yields the result of this lemma. □

Proof of Theorems 2 and 3. From the definition of ${}^*B_i(t)$ in (EC.26), we have $A_i({}^*B_i(t))\geq A_i(B_i(t))$. Since A_i is decreasing, this inequality implies ${}^*B_i(t)\leq B_i(t)$ for all $i=1,\dots,I$. Then it can be seen from (7) that $\sum_{i=1}^{I}{}^*B_i(t)\leq n$. This yields $\lim\limits_{t\to\infty}\sum_{i=1}{}^*B_i(t)=n$ by (EC.35). Then, we also have $\lim_{t\to\infty}\sum_{i=1}^{I}B_i(t)=n$ following from (7). Hence, $\lim_{t\to\infty}(B_i(t)-{}^*B_i(t))=0$ for all $i=1,\dots,I$. Thus we can conclude from (EC.26) that

$$\lim_{t\to\infty}\max_{1\leq k,l\leq I}|A_k(B_k(t))-A_l(B_l(t))|=0.$$

It then follows from Lemma EC.7 that $\lim\limits_{t\to\infty}B_i(t)=b_i^*$. This together with Proposition 1 yields $\lim\limits_{T\to\infty}J_T(\pi_{b^*})=\lim\limits_{T\to\infty}J_T(\pi_G)=J^*$. Till now we complete the proof. □

EC.2.4. Optimality of the Fixed Priority Policy

Proposition 2 shows that the fluid model given any fixed priority order converges to an equilibrium with a special form as (23). For concave holding cost functions and nondecreasing hazard rate functions, Theorem 4 states that the optimal scheduling policy must be in the family of the fixed priority policies. The proof is placed in the end of this subsection.

Recall from the definition of i_0 in (23), i_0 is the biggest number such that $\sum_{i=1}^{i_0-1}\lambda_i/\mu_i$ is strictly less than n, which implies that the traffic intensity of the first i_0-1 classes with high priorities are actually underloaded. Intuitively, their queue lengths should vanish after a finite time under a fixed priority scheduling. The following lemma verifies such a phenomena and claims that the first i_0-1 queues will become empty eventually.

Lemma EC.8. *Under Assumption 1, for any class $i\in\{1,\cdots,i_0-1\}$, where i_0 is given in* (23)*, the queue length vanishes after a finite time and the amount of customers in service converges to* (23)*. In other words, there exists a $T>0$ such that $Q_i(t)=0$ for all $t\geq T$ and $i\in\{1,\cdots,i_0-1\}$. And*

$$\lim_{t\to\infty}B_i(t)=\lambda_i/\mu_i \quad \textit{for all } i\in\{1,\cdots,i_0-1\}. \tag{EC.41}$$

Proof. We prove the result by induction.

Step 1: As a first step, we show this lemma holds for $i=1$. To prove this, we first show that $\liminf_{t\to\infty}B_1(t)\geq b_1=\frac{\lambda_1}{\mu_1}$. Suppose that $B_1(t)\leq\frac{\lambda_1}{\mu_1}-\delta$ for some $\delta>0$. Combining (1) with (EC.20) yields

$$B_1'(t)=K_1'(t)-D_1'(t)=\begin{cases}\sum_{i=1}^I\mu_iB_i(t)-\mu_1B_1(t) & \text{if } Q_1(t)>0,\\ [\sum_{i=1}^I\mu_iB_i(t)]\wedge\lambda_1-\mu_1B_1(t) & \text{if } Q_1(t)=0 \text{ and } \sum_{i=1}^I B_i(t)=n,\\ \lambda_1-\mu_1B_1(t) & \text{if } \sum_{i=1}^I B_i(t)<n.\end{cases}$$

Then, one can easily see from the above equation that $B_1'(t)\geq c>0$ for small constant c only depending on δ. Due to the arbitrariness of δ, the result $\liminf_{t\to\infty}B_1(t)\geq b_1=\frac{\lambda_1}{\mu_1}$ thus follows. Now for any $\epsilon>0$, we have $B_1(t)\geq b_1-\epsilon$ for all large t. This together with (2), (7) and the first entry of (EC.20) implies when $Q_1(t)>0$ we have

$$\begin{aligned}K_1'(t)=\sum_{i=1}^I D_i'(t)&\geq\mu_1B_1(t)+\mu_{\min}(n-B_1(t))\\&\geq\mu_1(b_1-\epsilon)+\mu_{\min}(n-b_1+\epsilon)\end{aligned}$$

$$\geq \mu_1 b_1 + \frac{1}{2}\mu_{\min}(n-b_1)$$

for small enough $\epsilon > 0$, where $\mu_{\min} = \min_{i=1,\ldots,I} \mu_i$. Thus, $Q_1'(t) \leq -\frac{1}{2}\mu_{\min}(n-b_1)$ whenever $Q_1(t) > 0$ from (4). Therefore there exists $t_1 > 0$ such that $Q_1(t) = 0$ for all $t \geq t_1$. Thus by Proposition 1 we have $\lim_{t\to\infty} B_1(t) = \lambda_1/\mu_1$.

Step 2: Suppose that Lemma EC.8 is true for all $i = 1, \cdots, k-1 \in \{1, \cdots, i_0 - 1\}$, i.e.,

$$\lim_{t\to\infty} B_i(t) = b_i \quad \text{for all } i = 1, \cdots, k-1. \tag{EC.42}$$

And there exists a $T_{k-1} > 0$ such that $\sum_{i=1}^{k-1} Q_i(t) = 0$ for all $t \geq T_{k-1}$. From this, we need to show that Lemma EC.8 continues to hold for $k \in \{1, \cdots, i_0 - 1\}$. Now by (4) we have $\sum_{i=1}^{k-1} K_i'(t) = \sum_{i=1}^{k-1} \lambda_i$ for all $t \geq T_{k-1}$. So from (2) and (EC.20), one can see that for all $t \geq T_{k-1}$,

$$K_k'(t) = \begin{cases} \sum_{i=1}^{I} \mu_i B_i(t) - \sum_{i=1}^{k-1} \lambda_i, & \text{if } Q_k(t) > 0, \\ \lambda_k \wedge \left(\sum_{i=1}^{I} \mu_i B_i(t) - \sum_{i=1}^{k-1} \lambda_i\right) & \text{if } Q_k(t) = 0 \text{ and } \sum_{i=1}^{I} B_i(t) = n, \\ \lambda_k & \text{if } \sum_{i=1}^{I} B_i(t) < n. \end{cases} \tag{EC.43}$$

By (1),

$$B_k'(t) = \begin{cases} \sum_{i=1}^{I} \mu_i B_i(t) - \mu_k B_k(t) - \sum_{i=1}^{k-1} \lambda_i, & \text{if } K_k'(t) = \sum_{i=1}^{I} \mu_i B_i(t) - \sum_{i=1}^{k-1} \lambda_i, \\ \lambda_k - \mu_k B_k(t), & \text{if } K_k'(t) = \lambda_k. \end{cases}$$

Similar to Step 1, we also show that $\liminf_{t\to\infty} B_k(t) \geq b_k = \frac{\lambda_k}{\mu_k}$. Suppose that $B_k(t) \leq \frac{\lambda_k}{\mu_k} - \delta$ for some $\delta > 0$. From (EC.42) and the above, one can conclude that $B_k'(t) \geq c > 0$ for a small constant c only depending on δ. As a consequence, we have $\liminf_{t\to\infty} B_k(t) \geq b_k = \frac{\lambda_k}{\mu_k}$. Note that $\mu_i b_i = \lambda_i$ for all $i \in \{1, \cdots, i_0 - 1\}$. Thus (EC.42) implies that for any $\epsilon > 0$

$$\sum_{i=1}^{k-1} \lambda_i - \epsilon \leq \sum_{i=1}^{k-1} \mu_i B_i(t) \leq \sum_{i=1}^{k-1} \lambda_i + \epsilon$$

for all large t. According to the above proved limit inferior of $B_k(t)$, for any $\epsilon' > 0$, we have $B_k(t) \geq b_k - \epsilon'$ for all large t. When $Q_k(t) > 0$, using (7) and (EC.43), we have

$$K_k'(t) = \sum_{i=1}^{I} \mu_i B_i(t) - \sum_{i=1}^{k-1} \lambda_i$$

$$\begin{aligned}
&\geq \sum_{i=1}^{k-1}(\mu_i B_i(t)-\lambda_i)+\mu_k B_k(t)+\mu_{\min}(n-\sum_{i=1}^{k}B_i(t)) \qquad \text{(EC.44)}\\
&\geq -\epsilon+(\mu_k-\mu_{\min})(b_k-\epsilon')+\mu_{\min}\left(n-\sum_{i=1}^{k-1}(b_i+\epsilon)\right)\\
&=\mu_k b_k+\mu_{\min}(n-\sum_{i=1}^{k}b_i)-\epsilon-\epsilon'\mu_k-(k-1)\epsilon'\mu_{\min}+\epsilon'\mu_{\min}\\
&\geq \mu_k b_k+\frac{1}{2}\mu_{\min}(n-\sum_{i=1}^{k}b_i)
\end{aligned}$$

for small enough $\epsilon,\epsilon'>0$. The above and (4) implies $Q_k'(t)\leq -\frac{1}{2}\mu_{\min}(n-\sum_{i=1}^{k}b_i)$ whenever $Q_k(t)>0$. Therefore, there exists a t_k such that $Q_k(t)=0$ for all $t\geq t_k$. Therefore, the result $\lim_{t\to\infty}B_k(t)=\lambda_k/\mu_k$ follows from Proposition 1. □

With Lemma EC.8, we now proceed with the proof of Proposition 2.

Proof of Proposition 2. Lemma EC.8 shows that the first i_0-1 classes with high priorities satisfy $\lim_{t\to\infty}B_i(t)=b_i$ and there exists a T such that $Q_i(t)=0$, $t\geq T$, for all $i\in\{1,\ldots,i_0-1\}$. And $\sum_{i=1}^{i_0-1}K_i'(t)=\sum_{i=1}^{i_0-1}\lambda_i$ for all $t\geq T$ from (4) and (6). Then it follows from (2) and (EC.20) that for all $t\geq T$,

$$K_{i_0}'(t)=\begin{cases}\sum_{i=1}^{I}\mu_i B_i(t)-\sum_{i=1}^{i_0-1}\lambda_i & \text{if } Q_{i_0}(t)>0,\\ \lambda_{i_0}\wedge\left(\sum_{i=1}^{I}\mu_i B_i(t)-\sum_{i=1}^{i_0-1}\lambda_i\right) & \text{if } Q_{i_0}(t)=0 \text{ and } \sum_{i=1}^{I}B_i(t)=n,\\ \lambda_{i_0} & \text{if } \sum_{i=1}^{I}B_i(t)<n.\end{cases} \qquad \text{(EC.45)}$$

In order to complete the proof of this theorem, a critical step is to prove $\lim_{t\to\infty}B_{i_0}(t)=b_{i_0}=n-\sum_{i=1}^{i_0-1}\frac{\lambda_i}{\mu_i}$, which is less than or equal to λ_{i_0}/μ_{i_0} according to the definition of i_0 in (23). Deducing from (7), (23) and (EC.41), there must be $\limsup_{t\to\infty}B_{i_0}(t)\leq b_{i_0}$. Then it suffices to show that $\liminf_{t\to\infty}B_{i_0}(t)\geq b_{i_0}$. To this end, we consider the following two cases.

Case 1: $i_0=I$. Suppose that $B_{i_0}(t)\leq b_{i_0}-\delta$ for some $\delta>0$. For large enough t, this could happen only when $\sum_{i=1}^{I}B_i(t)<n$. Otherwise, we have $\sum_{i=1}^{I}B_i(t)=n$. And by (7) and (EC.41) this causes $B_{i_0}(t)=n-\sum_{i=1}^{i_0-1}B_i(t)>b_{i_0}-\delta$ for all large enough t. So we just need to consider $\sum_{i=1}^{I}B_i(t)<n$. Then by (1) and (EC.45), $B_{i_0}'(t)=\lambda_{i_0}-\mu_{i_0}B_{i_0}(t)\geq\mu_{i_0}\delta$. This implies $\liminf_{t\to\infty}B_{i_0}(t)\geq b_{i_0}$. Combining the limit superior in the above, it immediately follows $\lim_{t\to\infty}B_{i_0}(t)=b_{i_0}$.

Case 2: $i_0 < I$. Deduce from (1) and (EC.45) that

$$B'_{i_0}(t) = \begin{cases} \sum_{i=1}^{I} \mu_i B_i(t) - \mu_{i_0} B_{i_0}(t) - \sum_{i=1}^{i_0-1} \lambda_i, & \text{if } K'_{i_0}(t) = \sum_{i=1}^{I} \mu_i B_i(t) - \sum_{i=1}^{i_0-1} \lambda_i, \\ \lambda_{i_0} - \mu_{i_0} B_{i_0}(t), & \text{if } K'_{i_0}(t) = \lambda_{i_0}. \end{cases}$$

Here we also suppose that $B_{i_0}(t) \le b_{i_0} - \delta$ for some $\delta > 0$. Together this with the above equation, one can find that if $K'_{i_0}(t) = \sum_{i=1}^{I} \mu_i B_i(t) - \sum_{i=1}^{i_0-1} \lambda_i$, then

$$\begin{aligned} B'_{i_0}(t) &= \sum_{i=1}^{i_0-1} [\mu_i B_i(t) - \lambda_i] + \sum_{i=i_0+1}^{I} \mu_i B_i(t) \\ &\ge \sum_{i=1}^{i_0-1} [\mu_i B_i(t) - \lambda_i] + \mu_{\min}(n - \sum_{i=1}^{i_0-1} B_i(t) - b_{i_0} + \delta) \\ &\ge \frac{1}{2} \mu_{\min} \delta, \end{aligned}$$

where the last inequality follows from (EC.41). If $K'_{i_0}(t) = \lambda_{i_0}$, then $B'_{i_0}(t) \ge \lambda_{i_0} - \mu_{i_0} b_{i_0} + \mu_{i_0} \delta \ge \mu_{i_0} \delta$. It then follows that $\liminf_{t\to\infty} B_{i_0}(t) \ge b_{i_0}$. As argued in the above this implies $\lim_{t\to\infty} B_{i_0}(t) = b_{i_0}$. Apparently, together this with (7) and (EC.41) yields $\lim_{t\to\infty} B_i(t) = 0$ for all $i = i_0 + 1, \cdots, I$. The convergence of queue length processes can be seen from Proposition 1. This completes the proof. □

Proof of Theorem 4. We claim that there exists a global minimum for which $0 < b_i < \lambda_i/\mu_i$ for at most one index i. From Lemma 1, the nonlinear programming (13) is a concave optimization problem if the cost functions C_i's are concave and the hazard rate functions h_i's are nondecreasing. Note that the constraint set is a convex set (acutally a convex polytope), then it follows that the optimization problem admits a global minimum at an extreme point, i.e., at one the vertices of this polytope. And at a vertex we have that $0 < b_i < \lambda_i/\mu_i$ for at most one index i. Corresponding to any optimal vertex, we can define an optimal fixed priority order. Then this theorem immediately follows from Propositions 1 and 2 (after re-ordering the class indices if needed). □

EC.3. Dynamic Programming Algorithm

This section is devoted to developing a dynamic programming (DP) algorithm to solve the Fractional 0-1 Knapsack Problem (29). It is easy to see that there exists a straightforward algorithm, especially when K is relatively small. According to each possible order of items, items are packed into the knapsack until the weight limit W is reached. Note that the last item packed might be divided. After evaluating all of the sequences, the optimal solution and the maximum value can be determined. However, such a brute-force algorithm is NP-hard. Fortunately, the DP algorithm of the classical 0-1 Knapsack Problem inspired us to develop a dynamic programming to solve it efficiently.

A DP Algorithm for the Fractional 0-1 Knapsack Problem. We determine how to optimally pack items into a knapsack, allowing at most one item to be divided, using a four-step procedure.

Step 1: Decompose the problem into subproblems.

In view of (29), any feasible solution contains at most one fractionally packed item. This suggests constructing a three-dimensional array $M[0..K, 0..W, 0..K]$, where the third dimension is used to track the fractionally packed item. For $1 \le k \le K$, $0 \le w \le W$ and $0 \le l \le K$, we consider the following two cases:

Case 1: $l = 0$. The entry $M[k, w, 0]$ stores the maximum rewarded value of items packed in their entirety from any subset of items $\{1, 2, \ldots, k\}$ with total weight at most w. The component 0 in $M[k, w, 0]$ indicates that there is no fractionally packed item.

Case 2: $l \neq 0$. The entry $M[k, w, l]$ stores the maximum rewarded value of the fractionally packed item l and the items packed in their entirety from any subset of items $\{1, 2, \cdots, k\} \setminus \{l\}$ with total weight at most w.

We also need the following initial setting for $k = 0$,

$$M[0, w, l] = \begin{cases} 0 & \text{if } l = 0, \\ V_l(w) & \text{if } l > 0 \text{ and } w_l > w, \\ -\infty & \text{if } l > 0 \text{ and } w_l \le w. \end{cases} \tag{EC.46}$$

The first entry means no item is packed in the knapsack. The second one implies that item l is fractionally packed with weight w since its full weight w_l exceeds the weight limit w. The third entry is illegal, since item l cannot be divided. Thus, we simply set the value to be $-\infty$. For the case with weight limit $w < 0$, which is also illegal, we set

$$M[k, w, l] = -\infty \quad \text{for all } w < 0 \text{ and } k, l \ge 0. \tag{EC.47}$$

Step 2: Recursively define the value of an optimal solution.

We use the above notations to define the rewarded value of an optimal solution recursively. Similar to the definition of $M[k, w, l]$, we recursively define it for two cases as well.

For $l = 0$, which means no item is fractionally packed, the optimal solution corresponding to $M[k, w, 0]$ is to either leave item k behind, in which case $M[k, w, 0] = M[k-1, w, 0]$, or pack item k, in which case $M[k, w, 0] = V_k(w_k) + M[k-1, w-w_k, 0]$ given $w_k \leq w$. Due to the penalty for a negative weight in (EC.47), we conclude that

$$M[k, w, 0] = \max\{M[k-1, w, 0], V_k(w_k) + M[k-1, w-w_k, 0]\} \tag{EC.48}$$

for all $1 \leq k \leq K$, $0 \leq w \leq W$. Actually, (EC.48) is exactly the recursive equation of the classical 0-1 Knapsack Problem (see §2.6 in Martello and Toth (1990)). For $l = 1, \ldots, K$, where item l is exactly the fractionally packed item, we can similarly derive

$$M[k, w, l] = \begin{cases} M[k-1, w, l] & \text{if } k = l, \\ \max\{M[k-1, w, l], V_k(w_k) + M[k-1, w-w_k, l]\} & \text{if } k \neq l. \end{cases} \tag{EC.49}$$

for all $1 \leq k \leq K$, $0 \leq w \leq W$, where the first entry means that item k has been fractionally packed, and thus it cannot also be packed in its entirety. The second entry relies on a similar explanation to that of (EC.48). Since this time item k is not the fractionally packed item, it can be either left behind or packed in the optimal solution corresponding to the maximum value $M[k, w, l]$.

We show in the proposition below that these recursions can indeed be described by a single recursive equation.

Proposition EC.1 (Recursive Equation). *The Fractional 0-1 Knapsack Problem* (29) *can be solved using dynamic programming. Namely, for any $l \in \{0, 1, \ldots, K\}$, we have the following recursive equation*

$$M[k, w, l] = \max\left\{M[k-1, w, l], V_k(w_k) + M[k-1, w-w_k, l] + \mathrm{Inf}\mathbf{1}_{\{k=l\}}\right\}, \tag{EC.50}$$

holds for all $k \in \{1, \ldots, K\}$ and $w \in \{0, 1, \ldots, W\}$, where $\mathrm{Inf} = -\infty$.

Proof. From the condition of this proposition, only $k \geq 1$ should be considered and $n = 0$ for the boundary condition has been given in (EC.46). Thus, it's easy to see that the recursions (EC.48) and (EC.49) can be expressed as a unified equation (EC.50). In order to prove (EC.50), we first consider a possible case $k = l$, which implies that item k is the fractionally added item. Then $M[k, w, l] = M[k-1, w, l]$ since in this case item k cannot be wholly taken. It remain to prove the case $k \neq l$. To compute $M[k, w, l]$ we note that there are only two choices for item k. If we leave the whole item k, then limited by the maximum weight w the maximum reward with the wholly added items taken from $\{1, 2, \cdots, k-1\}$ and the fractionally added being item l is $M[k-1, w, l]$. If instead we take the whole item k (only possible if $w \geq w_k$), then we gain $V_k(w_k)$ immediately, but consume w_k weight of our storage. Now the rest weight limit becomes $w - w_k$, then the maximum reward with the remaining items $\{1, 2, \cdots, k-1\}$ is $M[k-1, w-w_k, l]$. In all, we obtain $V_k(w_k) + M[k-1, w-w_k, l]$. Note that if $w < w_k$, then $M[k-1, w-w_k, l] = -\infty$ from (EC.47). So the recursion (EC.50) holds in both cases. □

Step 3: Compute the value of an optimal solution.

For any fixed $l \in \{0, 1 \ldots, K\}$, the above recursive equation (EC.50) suggests a two-dimensional recursive equation. In all, there are $K+1$ independent recursive equations. To reach our goal, we just need to recursively calculate $K+1$ two-dimensional recursions for $k \in \{1, \ldots, K\}$ and $w \in \{0, 1, \ldots, W\}$ based on the boundary conditions (EC.46) and (EC.47). Thus the running time of the dynamic programming algorithm is $O(K^2 W)$. Finally the optimal value of the Fraction 0-1 Knapsack Problem (29) is obtained as follows:

$$\max \sum_{k=1}^{K} V_k(y_k) = \max_{l \in \{0,1,\ldots,K\}} M[K, W, l]. \tag{EC.51}$$

Step 4: Construct an optimal solution.

From (EC.51), we find that $\mathit{Frac} := \arg\max_{l \in \{0,1,\ldots,K\}} M[K, M, l]$ is the index of the fractionally packed item of the optimal solution. The only remaining problem is to obtain the indices of the items that are packed in their entirety. To that end, we need one auxiliary three-dimensional array $\mathcal{T}[0..K, 0..W, 0..K]$ to be a Boolean array to find their indices. Each entry $\mathcal{T}[k, w, l]$ records whether item k is packed in its entirety in realizing the highest value $M[k, w, l]$. That is, $\mathcal{T}[k, w, l] = 1$ if item k is packed in its entirety and $\mathcal{T}[k, w, l] = 0$ otherwise. In the optimal solution, item K is packed in its entirety if $\mathcal{T}[K, W, \mathit{Frac}] = 1$. We can now repeat this argument for $\mathcal{T}[K-1, W-w_K, \mathit{Frac}]$. And item K is not packed in its entirety if $\mathcal{T}[K, W, \mathit{Frac}] = 0$. In this case, we can repeat the argument for $\mathcal{T}[K-1, W, \mathit{Frac}]$. Iterating the argument K times from item K downward to item 1 will give the indices of all items that are packed in their entirety.

Thus far we have identified the optimal value and the solution to (29). The step-by-step procedures are described in Algorithm 1.

Algorithm 1 The Fractional 0-1 Knapsack (Dynamic Programming)

```
procedure Initialization according to (EC.46)
procedure Recursively define values
for $k \leftarrow 1$ to $K$ do
    for $w \leftarrow 0$ to $W$ do
        for $l \leftarrow 0$ to $K$ do
            if $w_k \le w$ and $k \neq l$ and $M[k-1,w,l] < V_k(w_k) + M[k-1,w-w_k,l]$ then
                begin
                $M[k,w,l] \leftarrow V_k(w_k) + M[k-1,w-w_k,l]$
                $\mathcal{T}[k,w,l] \leftarrow 1$
                end
            else
                begin
                $M[k,w,l] \leftarrow M[k-1,w,l]$
                $\mathcal{T}[k,w,l] \leftarrow 0$
                end
procedure Search for the optimal value and the fractionally packed item
$Max \leftarrow M[K,W,0]$; $Frac \leftarrow 0$
for $k \leftarrow 1$ to $K$ do
    if $M[K,W,l] > Max$ then
        $Max \leftarrow M[K,W,l]$; $Frac \leftarrow l$
procedure Find indices of items packed in their entirety
$S \leftarrow W$
for $k \leftarrow K$ to $1$ do
    if $\mathcal{T}[k,S,Frac] = 1$ then
        $S \leftarrow S - w_k$; output $k$
```

Remark EC.1. To the best of our knowledge, the problem (29) was only studied in Burke et al. (2008). They also proposed an exact algorithm to solve that problem. The complexity of their approach is $O(UK^2W)$, where $U = \max_{k=1,\ldots,K} w_k$. The additional U is needed because they have to further calculate each possible value of the fractionally packed item. In contrast, the complexity our algorithm is only $O(K^2W)$ as shown in Step 3. Obviously, our proposed dynamic programming algorithm is more efficient. Note that the classical 0-1 Knapsack Problem needs $O(KW)$ time. More importantly, Propositions 3 and 4 reveal the internal connection between queueing and knapsack problems.