Universiteit Utrecht

Efficient flow scheduling in resource-sharing networks

Maaike Verloop

Date:	September 2005
Supervisors at CWI:	Prof. dr. ir. Sem Borst
	Dr. Rudesindo Núñez Queija
Supervisor at Utrecht University:	Prof. dr. ir. Erik Balder

Preface

This master thesis is the result of an eight-month internship at the Probability, Networks and Algorithms department of the Center for Mathematics and Computer Science (CWI) in Amsterdam. It completes my study "Wiskunde in Economie en Bedrijf" (WEB) at the Department of Mathematics at Utrecht University.

During these eight months I received a lot of support from my supervisors at CWI Sem Borst and Sindo Núñez Queija. They guided me through this research and were always willing to discuss and help me out with any problem. Moreover, I would like to thank them for reviewing this master thesis. Furthermore, I would like to thank Erik Balder for being my supervisor at Utrecht University.

I am grateful to CWI for offering me an eight-month internship. This internship gave me the opportunity to learn more about performance analysis of communication networks and to experience what doing research comes down to in practice. Besides that, the atmosphere at CWI was a very pleasant one to work in and I would like to thank the whole PNA2 group for their contribution to that.

Maaike Verloop
September 2005

Contents

Chapter 1

Introduction

1.1 Background and motivation

While there used to be separate networks for each type of communication, nowadays communication networks are being designed to support various applications on a common infrastructure. Applications such as page requests in web servers, data transfers in the Internet, telephony and video, can all make use of one underlying communication platform. This has the advantage that it provides the possibility of sharing the available resources such as link bandwidth and buffer space among the various classes of flows, allowing more efficient use of resources. Since Internet traffic is expected to continue to grow, the design of efficient service disciplines for deciding how resources are shared, is becoming more and more important for the delivery of large amounts of digital information. Besides efficient use of the resources, a related issue is the performance perceived by the users of the network. There is a strong need for the analysis of the performance of service disciplines in communication networks.

In situations where all flows only share one resource, size-based service disciplines (such as the Shortest Remaining Processing Time discipline (SRPT) or the Least Attained Service first discipline (LAS)) are popular mechanisms for improving the performance by favoring smaller service requests over larger ones. This approach is successful in systems that are work conserving, i.e. the capacity of a resource is fully used whenever there are flows to be served. However this is typically not the case in communication networks, because flows may use several resources. More precisely, in order to traverse the network from its origin to its destination a flow can require capacity from several resources simultaneously. A network with this property is called a resource-sharing network. The interaction between active flows, can cause non-work conserving behaviour in such a network: a resource with flows contending for its capacity, may not be fully used when flows require simultaneously service from another resource where capacity is more scarce. For example, consider the case of one flow requesting capacity at several resources simultaneously. Giving preference to cross traffic at one of the resources can cause the underutilization of the other resources.

Efficient flow scheduling disciplines in systems with non-work conserving behaviour have received little attention in the literature so far. In this thesis we present a mathematical analysis of efficient flow scheduling in a particular resource-sharing network.

1.2 Existing results in the literature

In this section we will briefly describe some existing results in the literature on single-server systems and resource-sharing networks.

1.2.1 Single server

In a single-server system, users require service from only one server or resource. The literature on efficient flow scheduling in single-server systems is very rich. This section briefly describes some optimality properties of scheduling disciplines.

In [8, 9] a single-server system is considered when information is available on the remaining service requirements. In [8] it is shown that SRPT minimizes sample-path wise the number of users in the system and consequently also the mean delay.

SRPT relies on the knowledge of the size of the service requirements, since it serves at each time the user with the shortest remaining service requirement. This information is generally not available. In [7] results are derived for policies that only use information regarding the attained service. It is proved that when there are multiple classes and each class has a service requirement distribution with decreasing failure rate (DFR), then the μ-rule is optimal in the sense that it stochastically minimizes the number of users in the system among all preemptive policies that use no knowledge of the remaining service requirements. (The μ-rule assigns priority to a user with the highest failure rate $r_i(s_i)$, which depends on its attained service s_i and its failure rate function $r_i(\cdot)$.) In case of exponentially distributed service requirements, this amounts to the Shortest Expected Remaining Processing Time first (SERPT) discipline. Moreover, if there is only one single class of users, all with the same DFR service requirements, then the μ-rule reduces to LAS. When instead there is only one single class of users, all with the same service requirement distribution with increasing failure rate (IFR), it is proved in [7] that any non-preemptive service discipline, and in particular the First Come First Served (FCFS) discipline, stochastically minimizes the number of users in the system at any point in time.

1.2.2 Resource-sharing networks

As mentioned above, in resource-sharing networks, also referred to as bandwidth-sharing networks, users require service from several shared resources simultaneously.

A popular class of policies studied in the literature for resource-sharing networks are the so-called weighted α-fair bandwidth-sharing policies as introduced in [5]. Such a policy allocates at time t the bandwidth or capacity of the resources to the flows, such that for a positive constant $\alpha \neq 1$, the utility function

$$\sum_i w_i n_i(t) \frac{s_i(t)^{1-\alpha}}{1-\alpha}$$

is maximized, subject to the capacity constraints. Here w_i is a weight corresponding to class i, $n_i(t)$ is the number of flows of class i at time t and $s_i(t)$ is the allocated capacity to each flow of class i at time t. These policies naturally favor flows that use fewer resources and therefore they achieve higher service parallelism. Moreover, the number of users is taken into account as well, so if one class becomes congested, this class will receive more capacity. Thus no class is strictly prioritized, which explains the adjective fair. Furthermore, through the weights the performance of a class can be influenced. The class of α-fair bandwidth-sharing policies includes several common fairness notions as special cases for various values of α, such as max-min fairness ($\alpha = \infty$), maximizing total throughput ($\alpha = 0$) and proportional fairness ($\alpha \to 1$).

In [2] it is proved that when the service requirements and the inter-arrival times are exponentially distributed, the network is stable under a weighted α-fair bandwidth-sharing policy if and only if the traffic load at each resource is smaller than its capacity. In [16] this is proved for a broader class of policies where a more general utility function is allowed.

Results with respect to the perceived performance of α-fair bandwidth-sharing policies in terms of mean transfer delays and flow throughput are scarce. Recently in [14, 15] the authors have focused on achieving good performance. They consider general resource-sharing networks and are interested in minimizing the bit transmission delay, i.e. delay/flow size, or equivalently, in maximizing the user-perceived throughput. They assume information on the flow sizes to be available and investigate how this information might be employed to maximize the user-perceived throughput. A size-based service discipline is proposed. This discipline uses as basis the weighted α-fair bandwidth-sharing policy, but with state-dependent weights. The per-user weights depend on the remaining service requirements. By simulations it is shown that these policies improve the user-perceived throughput compared to (weighted) α-fair bandwidth-sharing policies.

1.3 Goal and structure of the thesis

In Section 1.2 we noted that most of the research efforts on optimizing performance have focused on work-conserving systems consisting of a single server. However, for resource-sharing networks less is known. Most of the analysis is concerned with weighted α-fair bandwidth-sharing policies. For these disciplines the stability is guaranteed, but not much is known with respect to achieved performance.

The results for the single-server systems suggest that giving preference to smaller flows improves the overall performance of the system. However, giving size-based preference to flows in resource-sharing networks creates the possibility that certain resources are underutilized even when congestion builds up. This non-work conserving behaviour may render the network unstable, while for example an α-fair bandwidth-sharing policy would have ensured stability. If this is the case, these size-based service disciplines will certainly not provide good performance. Our first goal is therefore to derive stability conditions for service disciplines like SERPT, SRPT and LAS when applied in resource-sharing networks. This gives insights into whether straightforward extensions of size-based service disciplines provide good performance in resource-sharing networks as well.

Our second goal is to identify efficient service disciplines for resource-sharing networks, which minimize the number of flows in the network and thus the mean delay. These disciplines will leverage the advantages of achieving a high flow completion rate with preventing the system from persistent non-work conserving behaviour.

The thesis is organized as follows. In Chapter 2 we present a particular resource-sharing network, namely a linear network, which will be studied throughout the thesis. In Chapter 3 the stability conditions for size-based service disciplines are investigated, in particular for SRPT and LAS applied to the resource-sharing network. The contents of this chapter has been accepted for publication in Performance 2005, [12]. In Chapter 4 we identify policies which minimize the total number of users in the system, for certain choices of the parameters. In Chapter 5 we consider the remaining scenarios and partially characterize the optimal policies. Further, we examine a related fluid model for which we can completely derive the optimal policies.

Chapter 2

Model description

Our basic model is a linear network with L nodes. In general, each node i has a certain service rate, c_i, which can be shared among several users. In this thesis we assume that the nodes have a unit service rate, i.e. $c_i \equiv 1$ for all $i = 1 \ldots, L$. There are $L+1$ classes of users, where class-i users require service at node i only, $i = 1, \ldots, L$ and class-0 users require service at all L nodes simultaneously, see Figure 2.1.

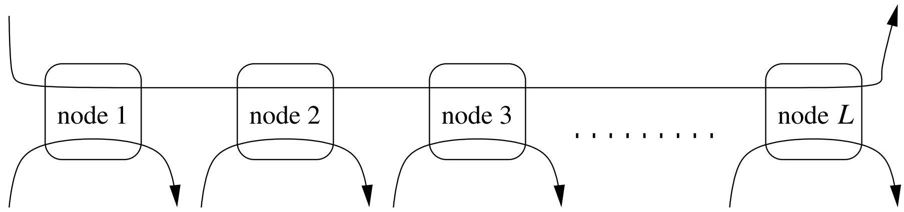

Figure 2.1: Linear network with L nodes.

The arrival processes of the various classes are independent. Class-i users arrive according to a Poisson process with parameter λ_i and have generally distributed service requirements B_i, with mean β_i, $i = 0, \ldots, L$. The traffic load of class i is given by $\rho_i = \lambda_i \beta_i$. At node i only classes 0 and i request service, therefore the traffic load at node i is $\rho_0 + \rho_i$.

The queue of class-i users is referred to as Q_i. Denote by the random variable $N_i(t)$ the number of class-i users in Q_i at time t. Define N_i as a random variable with the time-average distribution of $N_i(t)$ for $t \to \infty$, assuming it exists, i.e.

$$\mathbb{P}(N_i \leq n_i) = \lim_{t \to \infty} \frac{1}{t} \int_0^t \mathbb{P}(N_i(s) \leq n_i) \mathrm{d}s.$$

We define $\mathbf{N}(t) = (N_0(t), \ldots, N_L(t))$ and $N(t) = \sum_{i=0}^{L} N_i(t)$. Denote by $W_i(t)$ the workload of class i at time t, i.e. the sum of the remaining service requirements of all class-i users present in the system at time t.

A scheduling discipline or policy determines at each moment in time how the capacity of the nodes is distributed among the users, depending on the state of the system. At any time the allocated fraction of capacity to a user may be decreased or increased. To denote that the value of a variable results from a particular policy π, a π is placed in the superscript of the variable.

Notice that each user in the model represents a flow in the communication network described in Chapter 1. In reality, a flow consists of a sequence of packets traversing the network. Therefore the queue is a purely virtual entity in the sense that the users do not actually reside in physical queues, but rather keep the bulk of the backlogged work stored in their own buffers. In our model we abstract from packet-level details and view each flow as a continuous stream which arrives at once. The implementation of a flow-level scheduling discipline in a packet-based network involves major challenges. In this thesis we do not address such implementation issues, and assume that the service rates for the various users at the flow level can be adapted instantaneously as the state of the system changes.

Chapter 3

Stability of size-based service disciplines

In this chapter we explore the conditions for stability of the linear resource-sharing network under three different size-based service disciplines. These disciplines are straightforward extensions from the well-known disciplines SERPT, SRPT and LAS for single-server systems.

The service disciplines we consider rely on some underlying criterion which for every possible state of the network defines a priority ranking among all users, e.g. based on some class parameter (SERPT), remaining service requirement (SRPT), or amount of attained service (LAS). Since class-0 users require simultaneous service at all nodes, the priority ranking needs to be augmented with a further arbitration mechanism to arrive at the allocation of service rates among competing users. It can for instance happen that in node i a class-0 user has the highest priority, but in node $j \neq i$ there is a class-j user with a higher priority ranking. The class-j user is served, but it is left undetermined how the capacity of node i is allocated. We will distinguish between the following two options:

(i) 'strict' priority, which implies that the capacity in node i is left unused.

(ii) 'weak' priority, which implies that the capacity in node i is re-allocated to class i.

These two options may arise from different allocation schemes in networks. In a distributed rate allocation scheme each node allocates its capacity based on the users present at that node, while in a centralized rate allocation scheme all nodes are taken into consideration for determining the allocation of the capacity among all users in the network. Clearly, the former scheme requires less coordination between the nodes in determining the rate allocation among competing users. However, it does leave more capacity in the nodes unutilized compared to the centralized scheme.

The definition for stability is as follows.

Definition 3.0.1 (Stability) *Q_i is said to be stable if $\mathbb{P}(N_i = 0) > 0$. Node i, for $i = 1, \ldots, L$, is said to be stable if both Q_0 and Q_i are stable. The network is said to be stable if all nodes i, $i = 1, \ldots, L$ are stable.*

Since the capacity of a node, $c = 1$, must be adequate to cope with the total work load offered to it, $\rho_0 + \rho_i$, $\rho_0 + \rho_i < 1$ is obviously a necessary condition for node i to be stable. Thus $\rho_0 + \rho_i < 1$ for all $i = 1, \ldots, L$, are necessary conditions for the entire network to be stable. The results in [2] state that these conditions are in fact also sufficient for stability of α-fair bandwidth-sharing policies. We will refer to the necessary conditions as the 'standard' conditions.

Denote by $\sigma_i := \lim_{T\to\infty} \frac{1}{T}\int_{t=0}^{T} s_i(t)\mathrm{d}t$ the long-term average service rate of class i (assuming it exists), with $s_i(t)$ denoting the service rate allocated to class i at time t. Note that $\sigma_i = \rho_i$ when Q_i is stable. We have the identity relation $\sigma_0 + \sigma_i + u_i = 1$, where $u_i := \lim_{T\to\infty} \frac{1}{T}\int_{t=0}^{T} u_i(t)\mathrm{d}t$, with $u_i(t) := 1 - s_0(t) - s_i(t)$, denotes the long-term unused average service rate at node i. Noting that $u_i(t) = 1$ when $N_0(t) = 0, N_i(t) = 0$, we may write $u_i = w_i + \mathbb{P}(N_0 = 0, N_i = 0)$, where $w_i := \lim_{T\to\infty} \frac{1}{T}\int_{t=0}^{T} u_i(t)\mathbf{1}_{(N_0(t)>0 \text{ or } N_i(t)>0)}\mathrm{d}t$ stands for the long-term average 'wasted' service rate at node i. Here $\mathbf{1}_{(\cdot)}$ is the indicator function, that is

$$\mathbf{1}_{(A)} = \begin{cases} 1 & \text{if } A \text{ is true} \\ 0 & \text{if } A \text{ is false.} \end{cases}$$

Since $\mathbb{P}(N_0 = 0, N_i = 0) > 0$ implies stability of node i, it follows that node i is stable if the condition $\rho_0 + \rho_i + w_i < 1$ holds. Note that when the long-term average wasted service rates, w_i, are all equal to 0, the standard conditions are sufficient for stability of the network. In the linear network capacity can be wasted, hence the standard conditions are not necessarily sufficient. We have the following useful property.

Property 3.0.2 *(i) A sufficient condition for stability of the system is $\sum_{i=0}^{L} \rho_i < 1$, provided that at least one of the nodes operates at the full service rate whenever the network is non-empty.*
(ii) A sufficient condition for stability of Q_i, for $i = 1, \ldots, L$, is $\rho_0 + \rho_i < 1$, provided that node i operates at the full service rate whenever Q_i is non-empty.

Statement (i) follows from the fact that the total workload (of classes $i = 0, \ldots, L$), is dominated by that in a system where classes $i = 1, \ldots, L$ are never served at the same time. Statement (ii) deserves some more elaboration. Suppose that $\rho_0 + \rho_i < 1$, and hence $\sigma_0 + \sigma_i < 1$, while Q_i is not stable. Since Q_i is unstable, in the long run there are always class-i users and because node i operates at the full service rate whenever Q_i is non-empty, we have $w_i = 0$. Therefore $u_i = w_i + \mathbb{P}(N_0 = 0, N_i = 0) = 0$, so $\sigma_0 + \sigma_i = 1$, which is in contradiction with our initial supposition.

For the service disciplines considered in this thesis (SERPT, SRPT, LAS), the first property is always satisfied, although the second property only holds with certainty for the variants with weak priority.

The remainder of this chapter is organized as follows. In Section 3.1 we consider SERPT scheduling in the linear network for exponentially distributed service requirements and derive the stability conditions. In Sections 3.2 and 3.3 we investigate the stability conditions for SRPT and LAS, respectively. For certain limiting regimes we derive the exact stability conditions. In Section 3.4 we make some concluding remarks.

3.1 SERPT scheduling

In preparation for the analysis of the SRPT and LAS disciplines, we consider in this section the Shortest Expected Remaining Processing Time first (SERPT) discipline. In a single-server system, the policy SERPT serves at each point in time the user with the shortest expected remaining service requirement. In this section we assume that class-i users have exponentially distributed service requirements with mean $\beta_i = 1/\mu_i$. This implies that the expected remaining service requirement of a class-i user is $1/\mu_i$, independent of the attained service, since exponentially distributed random variables exhibit the memoryless property. Therefore, when $\mu_0 > \mu_i$, class-i users simply receive priority over class-0 users and when $\mu_i > \mu_0$ vice versa.

In Subsection 3.1.1 the stability conditions for the case of large class-0 users compared to classes $1, \ldots, L$ are derived. In Subsection 3.1.2 the stability conditions for the case of small class-0 users compared to classes $1, \ldots, L$ are derived. In Subsection 3.1.3 the stability conditions are discussed in a more general situation, where class-0 users are of intermediate size.
Note that the stability results that are derived in the remainder of this section, in fact hold for generally distributed service requirements and any static class-based priority discipline.

3.1.1 Large class-0 users

We first consider the case $\mu_0 < \mu_1, \ldots, \mu_L$, that is class-0 users have a larger mean (remaining) service requirement compared to classes $1, \ldots, L$. Since class-i users for $i = 1, \ldots, L$, always receive priority over class-0 users, they behave as in an isolated M/M/1 queue with class i only. Therefore, Q_i is stable if and only if $\rho_i < 1$, for $i = 1, \ldots, L$. Moreover, $N_1, \ldots, N_L$ are independent, so $\mathbb{P}(N_1 = n_1, \ldots, N_L = n_L) = (1 - \rho_1)\rho_1^{n_1} \cdots (1 - \rho_L)\rho_L^{n_L}$. Class-0 users can be served if and only if there are no class-i users, for $i = 1, \ldots, L$. Hence, Q_0 is stable if and only if

$$\rho_0 < \mathbb{P}(N_1 = 0, \ldots, N_L = 0) = \Pi_{i=1}^{L}(1 - \rho_i).$$

Note that the above condition is more stringent than the standard conditions (unless $\rho_i > 0$ for at most one $i = 1, \ldots, L$). In fact, the system can be unstable for arbitrarily low values of ρ_0 if the number of traversed nodes is large. In Sections 3.2 and 3.3 we show that the SRPT and LAS disciplines inherit these difficulties.

3.1.2 Small class-0 users

Now focus on the case $\mu_0 > \mu_1, \ldots, \mu_L$, that is class-0 users have smaller mean (remaining) service requirements compared to classes $1, \ldots, L$. Because class-0 users receive priority over all class-i users, $i = 1, \ldots, L$, class 0 behaves as in an M/M/1 queue with class 0 only. Hence Q_0 will be stable as long as $\rho_0 < 1$. Class-i users, for $i = 1, \ldots, L$, behave as in an isolated priority queue with classes 0 and i only. Such a system is work conserving, so Q_i will be stable when $\rho_0 + \rho_i < 1$.

3.1.3 Intermediate-size class-0 users

We now extend the model with class-i' users which only require service from node i, arrive according to a Poisson process of rate $\lambda_{i'}$, and have exponentially distributed service requirements with mean $1/\mu_{i'}$. Denote the traffic load of class i' by $\rho_{i'} := \lambda_{i'}/\mu_{i'}$, $i = 1, \ldots, L$.

We assume $\mu_i > \mu_0 > \mu_{i'}$ for all $i = 1, \ldots, L$, that is classes $i = 1, \ldots, L$ always receive priority and class 0 receives priority over all classes i', $i = 1, \ldots, L$. Hence class-i users, $i = 0, \ldots, L$, are not affected by the presence of class-i' users, $i = 1, \ldots, L$. It thus follows from the results in Subsection 3.1.1 that Q_i is stable if and only if $\rho_i < 1$ and that Q_0 is stable if and only if $\rho_0 < (1 - \rho_1) \cdots (1 - \rho_L)$.
In order to establish the stability condition for $Q_{i'}$, $i = 1, \ldots, L$, it is important to know whether we have weak or strict SERPT. Strict SERPT only allows a class-i' user to be served when there are no class-0 and class-i users in the system. In contrast, weak SERPT also allows a class-i' user to be served when there are class-0 users in the system which are however blocked from service by class-j users, $i \neq j$ and there are no class-i users present.

Weak SERPT

For weak SERPT, class-i' users can be served during the time that Q_i is empty. The fraction of time this happens is equal to $1 - \rho_i$. However, class-0 users may be served during this time as well. Thus, the stability condition for $Q_{i'}$ may be written as $\rho_{i'} < 1 - \rho_i - \sigma_0$, or equivalently $\rho_i + \rho_{i'} + \sigma_0 < 1$, where σ_0 denotes the fraction of time that class-0 users are served. In order to determine the value of σ_0, we need to distinguish whether Q_0 is stable or not, i.e. whether $\rho_0 < (1 - \rho_1) \cdots (1 - \rho_L)$ or not. If Q_0 is stable, then $\sigma_0 = \rho_0$, and thus the stability condition for $Q_{i'}$ becomes $\rho_{i'} < 1 - \rho_i - \rho_0$, or simply $\rho_0 + \rho_i + \rho_{i'} < 1$. If Q_0 is unstable, then $\sigma_0 = (1 - \rho_1) \cdots (1 - \rho_L)$, so that the stability condition for $Q_{i'}$ takes the form

$$\rho_{i'} < 1 - \rho_i - (1 - \rho_1) \cdots (1 - \rho_L) = (1 - \rho_i)(1 - \prod_{j \neq i} (1 - \rho_j)).$$

Strict SERPT

For strict SERPT, class-i' users can be served when there are no class-0 or i users in the network. The fraction of time this occurs is equal to $\mathbb{P}(N_0 = 0, N_i = 0)$. Therefore the stability condition for $Q_{i'}$ may be expressed as $\rho_{i'} < \mathbb{P}(N_0 = 0, N_i = 0)$ for $i = 1, \ldots, L$. In general no tractable expression appears to exist for $\mathbb{P}(N_0 = 0, N_i = 0)$.

3.2 SRPT scheduling

We now turn the attention to the Shortest Remaining Processing Time first (SRPT) discipline. In a single-server system, SRPT serves at each point in time the user with the shortest remaining service requirement. This discipline will be extended to our resource-sharing network in the following way. A class-0 user receives the total capacity of all nodes whenever it has the smallest remaining service requirement among all users. Otherwise, in case of weak SRPT the total capacity of node i is allocated to the class-i user with the smallest remaining service requirement. However, in case of strict SRPT the total capacity of node i is only granted to a class-i user, if this user has indeed the smallest remaining size among all class-0 and i users. Possible ties (which occur with non-zero probability in case of discrete service requirement distributions and deterministic services in particular) are assumed to be broken at random. The service requirements are assumed to be generally distributed.

For each class i, define x_i^* as the smallest value of x such that Q_i is unstable in a reference system where all class-i users with service requirement x or larger are denied access. It may be checked that, due to the mechanics of the SRPT discipline, in the original system all class-i users of smaller size than x_i^* eventually complete service and leave the system with probability one, whereas in the long run class-i users of size larger than or equal to x_i^* never complete service and stay in the system forever with non-zero probability. In fact, class-i users of strictly larger size than x_i^* will in the long run never even enter service. Note that Q_i is stable in the original system when $\mathbb{P}(B_i < x_i^*) = 1$ (so in particular when $x_i^* = \infty$ in case B_i has infinite support).
The relationship between the values of x_0^* and x_i^*, $i = 1, \ldots, L$, depends on whether the SRPT discipline used is weak or strict. Consider node i. Under both weak and strict SRPT, class-0 users with a given service requirement x cannot enter service until all the class-i users present at the time of their arrival with a service requirement smaller than or equal to x have completed service. Thus, if class-0 users with a given service requirement eventually leave the system, then class-i users with a smaller or equal service requirement must do so as well, which implies $x_0^* \leq x_i^*$. Moreover, for strict SRPT the reverse implication, $x_0^* \geq x_i^*$, also holds, since then also class-i users with a given service requirement x cannot enter service until all the class-0 users present at time of their arrival with a service requirement smaller than or equal to x have completed service. This is stated in the next proposition.

Proposition 3.2.1 *For weak SRPT, $x_0^* \leq x_i^*$, for $i = 1, \ldots, L$, and for strict SRPT, $x_0^* = x_i^*$, for $i = 1, \ldots, L$.*

Proof In order to avoid technicalities, we assume in this proof that the service requirement distributions of all classes have support everywhere. With minor modifications, the proof extends to distributions with zero density in some points by introducing 'fictitious' users and observing that B_0 and B_i cannot both have zero density in x_0^* and x_i^*, respectively.
Suppose $x_0^* > x_i^*$. For both weak and strict SRPT, we can reach a contradiction as follows. By definition of x_i^*, in the long run, when a class-i user arrives with service requirement s_i, $x_i^* < s_i < x_0^*$, it will never leave the system (and in fact never even enter service). Now suppose that subsequently a class-0 user arrives with service requirement s_0, $s_i < s_0 < x_0^*$. Because of SRPT, this user cannot enter service before the class-i user leaves the system. Since the latter never happens, the class-0 user never leaves the system either, which contradicts the assumption $s_0 < x_0^*$ and the definition of x_0^*. Thus, for both weak and strict SRPT, $x_0^* \leq x_i^*$, for $i = 1, \ldots, L$.
For strict SRPT, it may be shown along similar lines that $x_0^* < x_i^*$ leads to a contradiction. By definition of x_0^*, in the long run, when a class-0 user arrives with service requirement s_0, $x_0^* < s_0 < x_i^*$, it will never leave the system (and in fact never even enter service). Now suppose that subsequently a class-i user arrives with service requirement s_i, $s_0 \leq s_i < x_i^*$. With strict SRPT, this user cannot enter service before the class-0 user leaves the system. Since the latter never happens, the class-i user never leaves the system either, which contradicts the assumption $s_i < x_i^*$ and the definition of x_i^*. Thus, for strict SRPT, $x_0^* \geq x_i^*$, so that in fact $x_0^* = x_i^*$, for $i = 1, \ldots, L$. □

Denote by

$$\rho_i(x) := \lambda_i \mathbb{E}(B_i \mathbf{1}_{(B_i < x)}) = \lambda_i \int_0^{x^-} y \mathrm{d}B_i(y) \tag{3.1}$$

the traffic load of class i when all class-i users of size x or larger are rejected. It is important to note that users of size exactly x are excluded in this definition. The following observation can be made.

Observation 3.2.2 *For both weak and strict SRPT,* $\rho_0(x_0^*)+\rho_i(x_i^*) \leq 1$*, for* $i = 1, \ldots, L$*. For weak SRPT,* $\rho_0(x_0^*) + \rho_i(x_i^*) = 1$ *in case* Q_i *is unstable and* B_0 *and* B_i *have a continuous distribution.*

The observation follows from the properties that (i) $\sigma_0 + \sigma_i \leq 1$, $i = 1, \ldots, L$, with equality for weak SRPT in case Q_i is unstable, and (ii) $\sigma_j \geq \rho_j(x_j^*)$, $j = 0, \ldots, L$, for both weak and strict SRPT, with equality in case B_j has a continuous distribution.

Combining Proposition 3.2.1 and Observation 3.2.2, it follows that if $\rho_0 + \rho_i > 1$ for some $i = 1, \ldots, L$, then for strict SRPT it holds that $x_0^* = x_i^* < \infty$, so that both Q_0 and Q_i are unstable in case B_0 and B_i have infinite support. Unfortunately, the above results do not suffice to determine the exact values of x_i^* in general, since Proposition 3.2.1 gives equality only for strict SRPT, whereas the relation in Observation 3.2.2 only holds with equality for weak SRPT. In order to establish exact stability conditions, we need to impose some additional assumptions on the service requirement distributions, as will be done in the next subsections. In Subsections 3.2.1 and 3.2.2 we derive the stability conditions in the specific situations where class-0 users are large and small compared to classes $i = 1, \ldots, L$, respectively.

3.2.1 Large class-0 users

In this subsection we consider class-0 users with large service requirements compared to all other classes. First the condition for stability of Q_i, $i = 1, \ldots, L$, is established. After that, the stability conditions for class 0 are derived in a particular limiting regime.

Stability of Q_i, for $i = 1, \ldots, L$

Define $m_i := \inf\{x : B_i(x) > 0\}$ and $M_i := \sup\{x : B_i(x) < 1\}$ as the minimum and maximum values of the class-i service requirements, $i = 0, \ldots, L$. When class 0 has larger service requirements than all classes i, i.e., $m_0 > M_i$, for $i \neq 0$, a class-0 user can only enter service when there are no class-i users in the system. When a class-0 user is in service and a class-i user arrives, the service is preempted when the remaining service requirement of the class-0 user is larger than that of the arriving class-i user.
Evidently, $\rho_i < 1$ is a necessary condition for stability of Q_i, $i = 1, \ldots, L$, because otherwise Q_i would be unstable even in the absence of any class-0 users. But is this condition sufficient for stability of Q_i as well? The next proposition shows that this is indeed the case for weak SRPT when $m_0 > M_i$.

Proposition 3.2.3 *Suppose the service discipline is weak SRPT and* $m_0 > M_i$*. Then the condition* $\rho_i < 1$ *is sufficient for stability of* Q_i*.*

Proof As observed above, the fact that $m_0 > M_i$ implies that class i receives preemptive priority over class 0, unless a class-0 user has a smaller remaining service requirement than all class-i users (so at most M_i) and is being served. In the presence of this class-0 user, it depends on the other classes whether class 0 or class i is being served. But, as long as Q_i remains non-empty after the arrival of a new class-i user, it will be prevented from service for at most a duration M_i, since weak SRPT does not leave any capacity in node i unused when class i is present. When this class-0 user leaves the system, no new class-0 users are taken into service under SRPT as long as class i is present, since we assumed that $m_0 > M_i$. It follows that the number of class-i users behaves as in an isolated queue with class i only and random service interruptions whose total duration during each busy period is bounded by M_i. Lemma 3.2.4 implies that such a queue is stable for any $\rho_i < 1$. □

Lemma 3.2.4 *Consider an M/G/1 queue with traffic load ρ and with service interruptions. Assume that the total duration of the service interruptions in any contiguous period during which the queue is continuously backlogged is stochastically bounded by the random variable M. Further assume that M is independent of the arrival process and the service requirements and that $\mathbb{E}(M) < \infty$. Then for any work-conserving policy the queue is stable when $\rho < 1$.*

Proof In an ordinary M/G/1 queue without service interruptions we know that if $\rho < 1$, then $\mathbb{E}(BP) < \infty$, where BP is the random variable denoting the length of a busy period. Let the random variable C denote the length of a contiguous period during which the queue is continuously backlogged. With each user we can associate a sub-busy period during which that user is served, as well as users that arrived during that service time (not counting those that arrive when that service time is interrupted), those that arrived during the service of those users and so on. The period C can now be split into the following three components: the service interruptions, the sub-busy periods of the user that arrived at an empty system when there is no service interruption (this user may not be present) and the sub-busy periods of the users that arrived during a service interruption. The expected number of users that arrive while the service is interrupted is bounded by $\lambda\mathbb{E}(M)$. We can therefore write

$$\mathbb{E}(C) \leq \mathbb{E}(M) + (1 + \lambda\mathbb{E}(M))\mathbb{E}(BP) < \infty.$$

This implies $\mathbb{P}(N = 0) > 0$, which establishes the stability of the queue. □

The next proposition indicates that for strict SRPT the condition $\rho_i < 1$ is not sufficient in general for Q_i to be stable.

Proposition 3.2.5 *Suppose the service discipline is strict SRPT and $m_0 > M_i$. Then the condition for stability of Q_i is*

$$\rho_i < 1 \text{ and } \rho_j(M_i) < 1 \text{ for all } j \neq 0, i.$$

Proof We first prove that the above condition is sufficient. The fact that $m_0 > M_i$ implies that class i receives preemptive priority over class 0, and will be entitled to service, unless a class-0 user is present with a smaller remaining service requirement than all class-i users, regardless of whether it is being served or not. Although the service of such a class-0 user may repeatedly be interrupted by arriving class-j users, $j = 1, \ldots, L$, the latter users all have service requirements of at most M_i. At the moment that Q_i is empty and there is an arrival of a class-i user, the time that class i is prevented from service while Q_i remains non-empty is denoted by D_i. The number of class-i users behaves as in an isolated queue with class i only and random service interruptions with a total duration of D_i during each busy period. By Lemma 3.2.4, such a queue is stable for any $\rho_i < 1$, when D_i is finite with probability 1. Therefore it remains to be shown that D_i is finite with probability 1.
T_i is defined as the time it takes for a class-0 user with a remaining service requirement of $r_0 = M_i$, to receive the last M_i part of its service. D_i can be bounded from above by T_i, since class i only notices the class-0 user, when $r_0 \leq M_i$. Note that at the moment that the class-0 user is being

served and r_0 reaches the level M_i, because of SRPT, it is necessary that there are no other users present with remaining service requirement smaller than M_i, that is there are no class-$1,\dots,L$ users present.
Denote by $r_0(t)$ the smallest remaining service requirement of all the class-0 users present at time t. A class-0 user with remaining service requirement smaller than M_i is being served until a user of size smaller than $r_0(t_1)$ arrives at time t_1 (so it necessarily is of class i, $i = 1,\dots,L$). This user preempts the class-0 user. The class-0 user can resume its service when all newly arrived users with size not larger than $r_0(t_1)$ have left the system. This period is called a busy period of classes $1,\dots,L$.
After this busy period the class-0 user can enter service again, until at a certain time t_2 a user arrives with size smaller than $r_0(t_2)$ (such a user is necessarily of class i, $i = 1,\dots,L$). A new busy period starts of class-i users, $i = 1,\dots,L$, with sizes smaller than or equal to $r_0(t_2)$. This pattern repeats itself until the class-0 user has received its complete service and leaves the system.
Note that an upper bound for these busy periods is obtained when instead we look at the busy periods of users with size smaller than or equal to M_i. Since the class-0 user needs a total of M_i service, we can conclude that $(1-\rho_1(M_i))\cdots(1-\rho_L(M_i))\mathbb{E}(T_i) \leq M_i$. Hence, the class-0 user will eventually complete service, i.e. $\mathbb{E}(T_i) < \infty$, since $\rho_j(M_i) < 1$ for all $j = 1,\dots,L$. The fact that $\mathbb{E}(D_i) \leq \mathbb{E}(T_i) < \infty$ concludes the proof that the above condition is sufficient.

It remains to be shown that the above condition is necessary as well. $\rho_i < 1$ is clearly a necessary condition. To show that the second condition is necessary too, suppose it is not satisfied. Then $\rho_j(M_i) > 1$ for some j. Define $s_j = \sup\{s : \rho_j(s) \leq 1\}$, hence $s_j < M_i$. There is an b, $s_j < b \leq M_i$, such that there arrive class-j users with sizes in the interval $(s_j, b]$. For these class-j users the queue is unstable. Consider the last time epoch t^* that a class-j user arrives with size in the interval $(s_j, b]$ and in Q_j there are no users with size less than or equal to b. With a certain non-zero probability, there is a class-0 user in the system at time t^* with remaining service requirement r, with $s_j < b \leq r \leq M_i$. The service of this class-0 user will be preempted by the newly arrived class-j user at time t^*, and the service will never be resumed, since Q_j will never empty of class j users with size less than or equal to b again from time t^* onward. In the presence of this class-0 user, a possible present class-i user with remaining service requirement smaller than r can still be served, but after this class-i user has left, no class-i users with size greater than r will ever enter service again. Hence Q_i will grow indefinitely from time t^* onward. □

Stability of Q_0

We now turn to the stability of Q_0. To determine the sufficient condition for stability of Q_0, we will consider the network in a limiting regime, obtained by scaling the dynamics of some classes with a common parameter ϵ and passing $\epsilon \downarrow 0$.

We assume B_0 and B_i, $i = 1,\dots,L$ to be generally distributed. We will consider a sequence of systems, indexed by ϵ, where the class-i arrival rate in the ϵ-system is $\lambda_i^{(\epsilon)} := \lambda_i/\epsilon$ and the class-i service requirements are $B_i^{(\epsilon)} := \epsilon B_i$, for $i = 1,\dots,L$. Note that the traffic load of class i in the ϵ-system is $\rho_i^{(\epsilon)} = \frac{\lambda_i}{\epsilon}\epsilon\beta_i = \rho_i$, hence it is independent of ϵ. Furthermore, when we let $\epsilon \downarrow 0$, the class-i service requirements will become extremely small compared to class 0, so we are indeed in the situation of large class-0 users.

In the ϵ-system we will make a distinction between class-i users with original size smaller or larger than $\sqrt{\epsilon}$. The traffic load in the ϵ-system when all class-i users of size $\sqrt{\epsilon}$ or larger are rejected, is equal to $\rho_i^{(\epsilon)}(\sqrt{\epsilon})$ as defined in (3.1). In the limit this will contribute the total traffic load of class i, i.e. $\lim_{\epsilon\downarrow 0}\rho_i^{(\epsilon)}(\sqrt{\epsilon}) = \rho_i$. This follows from $\rho_i^{(\epsilon)}(\sqrt{\epsilon}) = \lambda_i^{(\epsilon)}\int_0^{\sqrt{\epsilon}} y\mathrm{d}B_i^{(\epsilon)}(y) = \lambda_i\int_0^{1/\sqrt{\epsilon}} z\mathrm{d}B_i(z)$, where we used substitution of variables with $y = \epsilon z$.
For $i = 1,\ldots,L$, we define $I_i^{\sqrt{\epsilon}}$ as a period where class-i users with original size smaller than $\sqrt{\epsilon}$ are not served in the ϵ-system. Denote by $N_i^{\sqrt{\epsilon}}(t)$ the number of class-i users in the ϵ-system with original size smaller than $\sqrt{\epsilon}$ present at time t. It is possible that $N_i^{\sqrt{\epsilon}}(t) > 0$ during a period $I_i^{\sqrt{\epsilon}}$, but that these class-i users are blocked from service by a class-0 user or a class-i user with an original size larger than $\sqrt{\epsilon}$, and a remaining service requirement smaller than $\sqrt{\epsilon}$. We define $A_i^{\sqrt{\epsilon}}$, for $i = 1,\ldots,L$, as a period where class-i users with original size smaller than $\sqrt{\epsilon}$ are served in the ϵ-system. Note that in this period the total capacity of node i is allocated to a class-i user with original size smaller than $\sqrt{\epsilon}$. With minor abuse of notation, $I_i^{\sqrt{\epsilon}}$ and $A_i^{\sqrt{\epsilon}}$ will also be used to indicate that the event occurs at an arbitrary time epoch. In these definitions we implicitly assume the stationary distributions of $I_i^{\sqrt{\epsilon}}$ and $A_i^{\sqrt{\epsilon}}$ to exist.

Note that class-i users with original size smaller than $\sqrt{\epsilon}$, for $i = 1,\ldots,L$, receive preemptive priority over all other users, unless one of the other users is being served with a remaining service requirement smaller than or equal to $\sqrt{\epsilon}$. The latter will occur at most a fraction of order $\sqrt{\epsilon}$ of the time, so class-i users with original size smaller than $\sqrt{\epsilon}$ will receive priority over the other users virtually all the time as $\epsilon\downarrow 0$. Thus, class i restricted to $\sqrt{\epsilon}$ will approximately behave as in an isolated queue with class i restricted to $\sqrt{\epsilon}$ only as $\epsilon\downarrow 0$. Moreover, since $\rho_i^{(\epsilon)}(\sqrt{\epsilon}) \to \rho_i$ and classes i for $i = 1,\ldots,L$ will behave roughly independently, this suggests that $\mathbb{P}(I_1^{\sqrt{\epsilon}},\ldots,I_L^{\sqrt{\epsilon}}) \to \Pi_{i=1}^L(1-\rho_i)$, as is confirmed by the next proposition.

Proposition 3.2.6 *For the network under consideration in the limiting regime, with the weak SRPT discipline and $\rho_0+\rho_i<1$ for $i = 1,\ldots,L$, it holds that*

$$\lim_{\epsilon\downarrow 0}\mathbb{P}(I_1^{\sqrt{\epsilon}},\ldots,I_L^{\sqrt{\epsilon}}) = \Pi_{i=1}^L(1-\rho_i).$$

Proof As described earlier, class i restricted to $\sqrt{\epsilon}$ behaves as in an isolated queue with class i only and random service interruptions. Let us now introduce a reference system with class i only and with the same arrival process and service requirements as in the original system, and class-i users with sizes larger than $\sqrt{\epsilon}$ are rejected. Define $\hat{A}_i^{\sqrt{\epsilon}}$ and $\hat{I}_i^{\sqrt{\epsilon}}$ as the active and idle periods of the reference system, respectively.
Note that $\mathbb{P}(A_i^{\sqrt{\epsilon}},\hat{A}_i^{\sqrt{\epsilon}}) = \mathbb{P}(\hat{A}_i^{\sqrt{\epsilon}}) - \mathbb{P}(I_i^{\sqrt{\epsilon}},\hat{A}_i^{\sqrt{\epsilon}})$ and $\mathbb{P}(A_i^{\sqrt{\epsilon}},\hat{A}_i^{\sqrt{\epsilon}}) = \mathbb{P}(A_i^{\sqrt{\epsilon}}) - \mathbb{P}(\hat{I}_i^{\sqrt{\epsilon}},A_i^{\sqrt{\epsilon}})$, so $\mathbb{P}(\hat{A}_i^{\sqrt{\epsilon}}) - \mathbb{P}(I_i^{\sqrt{\epsilon}},\hat{A}_i^{\sqrt{\epsilon}}) = \mathbb{P}(A_i^{\sqrt{\epsilon}}) - \mathbb{P}(\hat{I}_i^{\sqrt{\epsilon}},A_i^{\sqrt{\epsilon}})$. Property 3.0.2 (ii) gives that $Q_1,\ldots,Q_L$ are stable in the original system, hence $\mathbb{P}(A_i^{\sqrt{\epsilon}}) = \rho_i^{(\epsilon)}(\sqrt{\epsilon})$. Since $\mathbb{P}(\hat{A}_i^{\sqrt{\epsilon}}) = \rho_i^{(\epsilon)}(\sqrt{\epsilon}) = \mathbb{P}(A_i^{\sqrt{\epsilon}})$, it follows that $\mathbb{P}(I_i^{\sqrt{\epsilon}},\hat{A}_i^{\sqrt{\epsilon}}) = \mathbb{P}(\hat{I}_i^{\sqrt{\epsilon}},A_i^{\sqrt{\epsilon}})$.
We now proceed to derive an upper bound for the latter probabilities. Let us denote by $(I_i^{\sqrt{\epsilon}},\hat{A}_i^{\sqrt{\epsilon}})$ the event that the original system is in $I_i^{\sqrt{\epsilon}}$ and the reference system is in $\hat{A}_i^{\sqrt{\epsilon}}$. Observe that

when the reference system is active at time t, i.e. $\hat{N}_i^{\sqrt{\epsilon}}(t) > 0$, it holds that $N_i^{\sqrt{\epsilon}}(t) > 0$, because $N_i^{\sqrt{\epsilon}}(t) \geq \hat{N}_i^{\sqrt{\epsilon}}(t)$. Thus, in order for the event $(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}})$ to occur, it must be the case that $N_i^{\sqrt{\epsilon}}(t) > 0$, i.e. there is a class-$i$ user with original size smaller than $\sqrt{\epsilon}$, but it is not served. As noted earlier, this can only arise when a class-i user with original size greater than $\sqrt{\epsilon}$ or a class-0 user is present with a remaining service requirement smaller than $\sqrt{\epsilon}$.
Define $T_{[0,t]}(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}})$ as the amount of time that the event $(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}})$ occurs during the interval $[0,t]$. We have the bound $T_{[0,t]}(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}}) \leq \sum_{n=1}^{N_{[0,t]}} D_{i,n}^{\sqrt{\epsilon}}$, with $N_{[0,t]}$ denoting the number of class-0 users and class-i users with original size larger than $\sqrt{\epsilon}$, that are served during the interval $[0,t]$; the index n is used to denote the n-th such user, and $D_{i,n}^{\sqrt{\epsilon}}$ is the amount of time that class-i users with original size smaller than $\sqrt{\epsilon}$ are prevented from service because of user n. For weak SRPT, we have $D_{i,n}^{\sqrt{\epsilon}} \leq \sqrt{\epsilon}$, since no capacity is left unused in the presence of class-i users and user n needs only its last $\sqrt{\epsilon}$ amount of service.

Using the strong law of large numbers, we can conclude that for weak SRPT,

$$\mathbb{P}(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}}) = \lim_{t\to\infty} \frac{T_{[0,t]}(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}})}{t} \leq \lim_{t\to\infty} \frac{N_{[0,t]}}{t}\sqrt{\epsilon} \leq (\lambda_0 + \lambda_i^{(\epsilon)}\mathbb{P}(B_i^{(\epsilon)} > \sqrt{\epsilon}))\sqrt{\epsilon}. \tag{3.2}$$

Furthermore we have $\lim_{\epsilon\downarrow 0} \lambda_i^{(\epsilon)}\mathbb{P}(B_i^{(\epsilon)} > \sqrt{\epsilon})\sqrt{\epsilon} = \lim_{\epsilon\downarrow 0} \frac{\lambda_i}{\sqrt{\epsilon}}\mathbb{P}(B_i > \frac{1}{\sqrt{\epsilon}})$. It can be shown that when $\mathbb{E}(B_i) \leq \infty$, the last limit is equal to 0. Together with (3.2) we can conclude that

$$\lim_{\epsilon\downarrow 0} \mathbb{P}(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}}) = 0 \quad \text{for all } i = 1, \ldots, L. \tag{3.3}$$

Note that the complement of the event $(I_1^{\sqrt{\epsilon}}, \ldots, I_L^{\sqrt{\epsilon}})$ is $(A_1^{\sqrt{\epsilon}} \cup \ldots \cup A_L^{\sqrt{\epsilon}})$. Therefore we have

$$\begin{aligned} \mathbb{P}(I_1^{\sqrt{\epsilon}}, \ldots, I_L^{\sqrt{\epsilon}}) &= \mathbb{P}(\hat{I}_1^{\sqrt{\epsilon}}, \ldots, \hat{I}_L^{\sqrt{\epsilon}}) + \mathbb{P}(I_1^{\sqrt{\epsilon}}, \ldots, I_L^{\sqrt{\epsilon}}, (\hat{A}_1^{\sqrt{\epsilon}} \cup \ldots \cup \hat{A}_L^{\sqrt{\epsilon}})) \\ &\quad -\mathbb{P}(\hat{I}_1^{\sqrt{\epsilon}}, \ldots, \hat{I}_L^{\sqrt{\epsilon}}, (A_1^{\sqrt{\epsilon}} \cup \ldots \cup A_L^{\sqrt{\epsilon}})). \end{aligned} \tag{3.4}$$

Since $\mathbb{P}(I_1^{\sqrt{\epsilon}}, \ldots, I_L^{\sqrt{\epsilon}}, (\hat{A}_1^{\sqrt{\epsilon}} \cup \ldots \cup \hat{A}_L^{\sqrt{\epsilon}})) \leq \sum_{i=1}^L \mathbb{P}(I_1^{\sqrt{\epsilon}}, \ldots, I_L^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}}) \leq \sum_{i=1}^L \mathbb{P}(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}})$, by (3.3) we can conclude that $\lim_{\epsilon\downarrow 0} \mathbb{P}(I_1^{\sqrt{\epsilon}}, \ldots, I_L^{\sqrt{\epsilon}}, (\hat{A}_1^{\sqrt{\epsilon}} \cup \ldots \cup \hat{A}_L^{\sqrt{\epsilon}})) = 0$. By symmetry this also holds for $\mathbb{P}(\hat{I}_1^{\sqrt{\epsilon}}, \ldots, \hat{I}_L^{\sqrt{\epsilon}}, (A_1^{\sqrt{\epsilon}} \cup \ldots \cup A_L^{\sqrt{\epsilon}}))$, since $\mathbb{P}(I_i^{\sqrt{\epsilon}}, \hat{A}_i^{\sqrt{\epsilon}}) = \mathbb{P}(\hat{I}_i^{\sqrt{\epsilon}}, A_i^{\sqrt{\epsilon}})$. Together with (3.4) this gives $\lim_{\epsilon\downarrow 0} \mathbb{P}(I_1^{\sqrt{\epsilon}}, \ldots, I_L^{\sqrt{\epsilon}}) = \lim_{\epsilon\downarrow 0} \mathbb{P}(\hat{I}_1^{\sqrt{\epsilon}}, \ldots, \hat{I}_L^{\sqrt{\epsilon}})$. The reference systems are independent and $\mathbb{P}(\hat{I}_i^{\sqrt{\epsilon}}) = 1 - \rho_i^{(\epsilon)}(\sqrt{\epsilon})$. Since $\rho_i^{(\epsilon)}(\sqrt{\epsilon}) \to \rho_i$ we can conclude that $\lim_{\epsilon\downarrow 0} \mathbb{P}(I_1^{\sqrt{\epsilon}}, \ldots, I_L^{\sqrt{\epsilon}}) = \Pi_{i=1}^L (1 - \rho_i)$. □

From Proposition 3.2.6 we can now derive the stability condition for ϵ small enough in an ϵ-system.

Corollary 3.2.7 *For the network under consideration in the limiting regime, with the weak SRPT discipline, we have (i) if $\rho_0 < \Pi_{i=1}^L (1 - \rho_i)$, then there exists an $\bar{\epsilon}$ such that Q_0 is stable in the ϵ-system for every $\epsilon < \bar{\epsilon}$.*
(ii) Conversely, if $\rho_0 > \Pi_{i=1}^L (1 - \rho_i)$, then there exists an $\bar{\epsilon}$ such that Q_0 is unstable in the ϵ-system for every $\epsilon < \bar{\epsilon}$.

Proof For (i), observe that if the traffic load of class-0 users and class-i users with original size larger than or equal to $\sqrt{\epsilon}$, $\rho_0 + \sum_{i=1}^{L}(\rho_i - \rho_i^{(\epsilon)}(\sqrt{\epsilon}))$, is smaller than the fraction of time that the system is not working on class-$1,\ldots,L$ users with original size smaller than $\sqrt{\epsilon}$, $\mathbb{P}(I_1^{\sqrt{\epsilon}},\ldots,I_L^{\sqrt{\epsilon}})$, then Q_0 is stable, since at least one of the nodes works at full rate whenever the system is non-empty. Therefore $\rho_0 + \sum_{i=1}^{L}(\rho_i - \rho_i^{(\epsilon)}(\sqrt{\epsilon})) < \mathbb{P}(I_1^{\sqrt{\epsilon}},\ldots,I_L^{\sqrt{\epsilon}})$ is a sufficient condition for Q_0 to be stable. Since $\rho_i^{(\epsilon)}(\sqrt{\epsilon}) \to \rho_i$, for $i = 1,\ldots,L$, Proposition 3.2.6 implies that for any $\rho_0 < \Pi_{i=1}^{L}(1-\rho_i)$ there exists an $\bar{\epsilon}$ such that Q_0 is stable in the ϵ-system for every $\epsilon < \bar{\epsilon}$.
Conversely, for (ii), for any $\rho_0 > \Pi_{i=1}^{L}(1-\rho_i)$ there exists an $\bar{\epsilon}$ such that $\rho_0 > \mathbb{P}(I_1^{\sqrt{\epsilon}},\ldots,I_L^{\sqrt{\epsilon}})$, for all $\epsilon \leq \bar{\epsilon}$. Note that $\mathbb{P}(I_1^{\sqrt{\epsilon}},\ldots,I_L^{\sqrt{\epsilon}}) \geq \mathbb{P}(I_1,\ldots,I_L)$. Since $\rho_0 > \mathbb{P}(I_1,\ldots,I_L)$ implies that Q_0 is unstable, the proof of (ii) is completed. □

3.2.2 Small class-0 users

We now turn the attention to the case where class-0 users have small service requirements compared to all other classes. In particular, we consider the service requirement distributions such that each class-0 user has original service requirement smaller than that of a class-i user, i.e. $M_0 < m_i$, for all $i = 1,\ldots,L$. Again we are interested in conditions on the traffic loads that guarantee a stable network. We will first consider the stability of Q_0 and later that of Q_i, $i = 1,\ldots,L$.

Stability of Q_0

In contrast to Corollary 3.2.7, the minimal condition $\rho_0 < 1$ can in this case already be *sufficient* for stability of Q_0. Moreover, for strict SRPT it is the stability condition for Q_0.

Observation 3.2.8 *When the service discipline is strict SRPT and $M_0 < m_i$ for all $i = 1,\ldots,L$, then the condition for stability of Q_0 is $\rho_0 < 1$.*

This may be deduced as follows. The fact that $M_0 < m_i$ implies that class 0 receives preemptive priority over class i, and will be entitled to service, unless a class-i user, for some $i = 1,\ldots,L$, has a smaller remaining service requirement than all class-0 users (so at most M_0). Class 0 has to wait until those users with remaining service requirement smaller than all class-0 users have left the network. Since it is strict SRPT, no new class-$1,\ldots,L$ users are taken into service. Thus, as long as Q_0 remains non-empty after the arrival of a new class-0 user, it will be prevented from service for at most a period M_0. From this it follows that the number of class-0 users behaves as in an isolated queue with class 0 only and random service interruptions whose total duration during each busy period is bounded by M_0. By Lemma 3.2.4, such a queue is stable for any $\rho_0 < 1$.

Under weak SRPT, $\rho_0 < 1$ is not necessarily sufficient for stability of Q_0. However, first we will illustrate a situation in which it is a sufficient condition. For that purpose we consider deterministic service requirements and $L = 2$ nodes.

Proposition 3.2.9 *Assume class i has a deterministic service requirement d_i, $i = 1, 2$, with $d_1 \neq d_2$ and $d_0 < d_1, d_2$, or $d_1 = d_2 > 2d_0$. For the network under consideration with the weak SRPT discipline, Q_0 is stable if and only if $\rho_0 < 1$.*

Proof The fact that $d_0 < d_i$ implies that class 0 receives preemptive priority over class i, and will be entitled to service, unless a class-i user, for some $i = 1, \ldots, L$, has a smaller remaining service requirement than d_0. Although class-$1, \ldots, L$ users may continue to be served for a while, the delay incurred by a newly arrived class-0 user is bounded as will be shown below. Thus, as long as Q_0 remains non-empty after the arrival of a new class-0 user, it will be prevented from service for at most a bounded period. Now it follows that the number of class-0 users behaves as in an isolated queue with class 0 only and random service interruptions whose total duration during each busy period is bounded. By Lemma 3.2.4, such a queue is stable for any $\rho_0 < 1$.
It remains to be shown that the delay incurred by a newly arrived class-0 user is bounded. Suppose that class 0 could be prevented from entering service indefinitely. Then at a certain point in time we have for example a class-1 user with a remaining service requirement $r_1 < d_0$ as well as class-0 and class-2 users of which none have received any service. Because of weak SRPT, the class-1 and class-2 users are served. When the class-1 user leaves the system, the class-2 user has remaining service requirement of $r_2 = d_2 - r_1$. When r_2 is smaller than d_0, this class-2 user is served and because of weak SRPT, a class-1 user also receives service. In order for this to repeat indefinitely, it is necessary that

$$r_1 < d_0, \quad 0 < d_2 - r_1 < d_0, \quad 0 < d_1 - d_2 + r_1 < d_0,$$

$$2d_2 - d_1 - r_1 < d_0, \quad 0 < 2d_1 - 2d_2 + r_1 < d_0, \ldots$$

or equivalently

$$k(d_1 - d_2) + r_1 < d_0 \text{ and } k(d_2 - d_1) + d_2 - r_1 < d_0, \quad \forall\, k \geq 0. \tag{3.5}$$

When $d_1 \neq d_2$, we can choose a K, such that for all $r_1 < d_0$ there exists a $k = k(r_1) < K$ for which (3.5) is not satisfied. When $d_1 = d_2 > 2d_0$, we may choose $K = 1$. We can conclude that at some point in time all class-1 and class-2 users have remaining service requirements greater than d_0, so a class-0 user can enter service. The delay for class 0 is therefore bounded by $(K+1)d_0$, independent of r_1. □

In general, $\rho_0 < 1$ is not a sufficient condition for stability of Q_0 under weak SRPT, as may be illustrated again with deterministic service requirements and $L = 2$ nodes. Take $d_1 = d_2 = d$ with $d_0 < d < 2d_0$ and assume that Q_1 and Q_2 are both unstable. In that case, the staggered service pattern of class-1 and class-2 users described in the proof of the above proposition may in fact replicate itself ad infinitum and class 0 can never return to service. Hence, Q_0 may also become unstable with non-zero probability. If Q_1 or Q_2 is stable, which is the case if $\rho_0 + \rho_1 < 1$ or $\rho_0 + \rho_2 < 1$, then with probability 1 the above cycle cannot repeat indefinitely, and it may in fact be checked that Q_0 is stable.

Let us now go back to general service requirements with $M_0 < m_i$, for $i = 1, \ldots, L$. As suggested by Proposition 3.2.9 the condition $\rho_0 < 1$ is sufficient for Q_0 to be stable only for specific service requirement distributions, where class 0 can only be prevented from service for a finite random amount of time. In general it is not, since a possible staggered service pattern of certain unstable Q_i's, $i = 1, \ldots, L$, may replicate itself ad infinitum. However such a scenario will never happen if for all $i = 1, \ldots, L$, Q_i is stable. In that case, when there are no class-0 users left, classes $1, \ldots, L$ start service. Class i is stable, therefore class i empties in a finite random amount of time. At this moment classes $1, \ldots, L$ behave independently of each other, therefore the random amount of time needed to empty the network of classes $i = 1, \ldots, L$ is finite and hence Q_0 is stable. Since weak SRPT satisfies Property 3.0.2, it follows that $\rho_0 + \rho_i < 1$ is a sufficient condition for Q_i to be stable. Therefore we have that the network is stable if $\rho_0 + \rho_i < 1$ for all $i = 1, \ldots, L$, as is stated in the next observation.

Observation 3.2.10 *Suppose the service discipline is weak SRPT and $M_0 < m_i$ for all $i = 1, \ldots, L$. Then the condition for stability of the network is $\rho_0 + \rho_i < 1$, for all $i = 1, \ldots, L$.*

Stability of Q_i, for $i = 1, \ldots, L$

We will now investigate the conditions for stability of Q_i.
Under weak SRPT, it follows from Property 3.0.2 that $\rho_0 + \rho_i < 1$ is a sufficient condition for stability of Q_i.
Under strict SRPT, $\rho_0 + \rho_i < 1$ will in general not be sufficient for stability of Q_i, $i = 1, \ldots, L$. We will show this by considering again deterministic service requirements and $L = 2$ nodes. As noted earlier, if $\rho_0 + \rho_i + w_i < 1$, then node i, and hence Q_i, is stable. Here the long-term average wasted service rate, w_i, is precisely the fraction of time that there are no class-i users with remaining service requirement smaller than d_0 and there are new class-0 users in the system which cannot be served because of the presence of a class-j user with a remaining service requirement smaller than d_0, $j \neq 0, i$. During this time no class-0 or class-i users can be served. Observe that $w_i > 0$ since there is a non-zero probability that an arriving class-0 user finds Q_0 empty and a class-j user in service with a remaining service requirement smaller than d_0, $j \neq 0, i$. An explicit expression for w_i appears hard to find.

3.3 LAS scheduling

In this section we consider the Least Attained Service first (LAS) discipline. In a single-server system, the policy LAS serves, in a processor-sharing way, those users in the system that have attained the least amount of service. Hence, when a new user arrives, it preempts the users in service and the newly arrived user receives the entire service capacity. This continues until one of the following events happens: it departs, a new user arrives, or it has obtained an amount of service equal to that received by the users preempted on arrival. In our linear resource-sharing network, LAS will be applied in the following fashion, where again we distinguish between the two variants: weak and strict. In each node, the users with the least attained service are granted the right to an equal share of the capacity at that node. A class-0 user requires in all nodes the same fraction, so class-0 users only receive the minimum of the granted shares at the nodes. This may leave some capacity unused at the nodes with the larger relative proportion of class-0 users. In case of weak LAS, the unused capacity is re-allocated to the other class at that node (if there are users of that class). In case of strict LAS, the unused capacity is simply lost. For the exposition, the assumption below will sometimes be convenient.

Assumption 3.3.1 *The service requirements of all classes have general* continuous *distributions.*

The subsequent analysis is facilitated by a particular property of LAS: the users with a total service requirement x are not influenced by users that have received more than x in service. It will therefore sometimes be convenient to consider the network or system where the service requirements of class

i are truncated at x, i.e. their service requirements are given by $\min(B_i, x)$. We define the following quantities, which we refer to as truncated loads:

$$\tilde{\rho}_i(x) := \lambda_i \mathbb{E}(\min(B_i, x)) = \lambda_i \int_0^{x^-} y \mathrm{d}B_i(y) + \lambda_i x \mathbb{P}(B_i \geq x) = \rho_i(x) + \lambda_i x \mathbb{P}(B_i \geq x),$$

where $\rho_i(x)$ was previously defined in (3.1). Thus, $\tilde{\rho}_i(x)$ represents the load due to class-i users *truncated* at size x (users larger than or equal to x contribute an amount x, rather than zero as in $\rho_i(x)$). We call the system obtained by truncating the sizes of class-i users at x_i, $i = 0, \ldots, L$, the $(x_0, \ldots, x_L)$-truncated system. If $x_0 = \ldots = x_L = x$ we simply refer to the "x-truncated" system. The ∞-truncated system corresponds to the original one.

Property 3.3.2 *From the perspective of users of size x, the system dynamics are identical to those of the x-truncated system. In addition, if $\mathbb{P}(B_0 \leq \bar{x}_0) = 1$, then from the perspective of class-j users, $j = 1, \ldots, L$, the system behaves identically to the $(\infty, \bar{x}_0, \ldots, \bar{x}_0, \infty, \bar{x}_0, \ldots, \bar{x}_0)$-truncated system, with ∞ in the first and $j+1$-th component.*

While the first claim is immediate from the arguments above, the second statement deserves some elaboration. A class-i user with attained service x_i, $i \neq j$, can only have influence on the service given to class-j users when there is a class-0 user present with attained service larger than or equal to x_i. If no class-0 user is larger than $\bar{x}_0$, then class-i users with attained service larger than $\bar{x}_0$ have therefore no influence on the class-j users.

Analogous to SRPT, we will again define an x_i^* that will indicate which class-i users experience a stable system. Note however that these values do not correspond to those for SRPT.
By choosing x small enough, we can ensure that $\sum_{i=0}^{L} \tilde{\rho}_i(x) < 1$, which implies by Property 3.0.2 (i) that there exists a stable x-truncated system. Furthermore, stability is monotone with respect to truncation, that is if $(x_0, \ldots, x_L) \geq (y_0, \ldots, y_L)$ component-wise and Q_i is stable in the $(x_0, \ldots, x_L)$-truncated system, then Q_i is also stable in the $(y_0, \ldots, y_L)$-truncated system. Under Assumption 3.3.1 we can therefore define

$$x_i^* := \sup\{x : Q_i \text{ is stable in the } x\text{−truncated system}\}.$$

It follows from Property 3.3.2 that class-i users of size at most x experience a stable system if and only if Q_i is stable in the x-truncated system, for $i = 0, \ldots, L$. Hence all class-i users of size smaller than x_i^* complete service in the original system. In the long run, class-i users of size larger than x_i^* only receive an amount of service equal to x_i^*. Note that Q_i is stable in the original system if $\mathbb{P}(B_i < x_i^*) = 1$.

The relationship between the values of x_0^* and x_i^*, $i = 1, \ldots, L$, depends on whether the LAS discipline used is weak or strict. Under both weak and strict LAS, class-0 users of a given size x cannot complete service until all class-i users of original size smaller than or equal to x have been cleared from the system. Thus, if the class-0 users of a given size eventually leave the system, then class-i users must do so as well, which implies $x_0^* \leq x_i^*$. Moreover, for strict LAS the reverse implication, $x_0^* \geq x_i^*$, also holds, since then class-i users of a given size x cannot complete service until all class-0 users of original size smaller than x have been cleared from the network. This results in the following proposition.

Proposition 3.3.3 *Suppose that Assumption 3.3.1 is satisfied and that all service requirement distributions have infinite supports, i.e.* $\mathbb{P}(B_i > x) > 0$ *for all* x. *For weak LAS it holds that* $x_0^* \leq x_i^*$, *for* $i = 1, \ldots, L$ *and for strict LAS it holds that* $x_0^* = x_i^*$, *for* $i = 1, \ldots, L$.

Proof The idea of the proof is similar to that of Proposition 3.2.1. Suppose $x_0^* > x_i^*$ for some $i = 1, \ldots, L$. For both weak and strict LAS, we can reach a contradiction as follows. By definition of x_i^*, in the long run, when a class-i user arrives of size s_i, $x_i^* < s_i < x_0^*$, it will never leave the system. Now suppose that subsequently a class-0 user arrives of size s_0, $s_i < s_0 < x_0^*$. Because of LAS, this user cannot complete service until the class-i user does. Since the latter never happens, the class-0 user never leaves the system either, which contradicts the assumption $s_0 < x_0^*$ and the definition of x_0^*. Thus, for both weak and strict LAS, $x_0^* \leq x_i^*$.
For strict LAS, it may be shown using similar arguments that $x_0^* < x_i^*$ leads to a contradiction. Thus for strict LAS we even have $x_0^* = x_i^*$. □

We are interested in the stability conditions under LAS scheduling. In the next two subsections exact stability conditions for the network are derived in limiting regimes. Subsections 3.3.1 and 3.3.2 consider limiting regimes where the sizes of class-0 users become extremely large and small, respectively.

3.3.1 Large class-0 users

In this subsection we investigate the stability condition under LAS scheduling in case class-0 users have large service requirements compared to those of class-i users, $i = 1, \ldots, L$. We do this by considering a limiting regime. As before, define a sequence of systems indexed by ϵ and let $\epsilon \downarrow 0$. Class-0 users arrive according to a Poisson process of rate $\lambda_0^{(\epsilon)} := \epsilon\lambda_0$ and sizes are distributed as $B_0^{(\epsilon)} := B_0/\epsilon$. As we let $\epsilon \downarrow 0$, the service requirements of class 0 become extremely large compared to those of classes $1, \ldots, L$, but the traffic load of class 0, ρ_0, remains constant. The next proposition shows that for an ϵ-system with ϵ small enough, to obtain a stable network it is necessary that $\rho_0 \leq \prod_{i=1}^{L}(1 - \rho_i)$.

Proposition 3.3.4 *Assume the service discipline is either weak or strict LAS. If there exists an* $\bar{\epsilon}$ *such that* Q_0 *is stable in the* ϵ*-system, for all* $0 < \epsilon < \bar{\epsilon}$, *then it must be that* $\rho_0 \leq \prod_{i=1}^{L}(1 - \rho_i)$.

Proof Let us focus on the ϵ-system. If Q_0 is stable, there must be sufficient capacity to serve all its traffic. In particular, this must be true for traffic due to class-0 users of total size larger than $h > 0$. Under LAS scheduling, once these users have received an amount of service equal to h, they can only be served when no users are present with attained service less than h. The traffic load belonging to the class-0 users that have attained already h in service (that is, not counting the first h amount of service given to them) is equal to $\rho_0 - \tilde{\rho}_0^{(\epsilon)}(h)$. Denote by $N_i^{(\epsilon,h)}$ the number of class-i users with attained service less than h. Since the traffic load must not be larger than the fraction of time in which it can be served, we necessarily have the following inequality:

$$\rho_0 - \tilde{\rho}_0^{(\epsilon)}(h) \leq \mathbb{P}(N_1^{(\epsilon,h)} = N_2^{(\epsilon,h)} = \cdots = N_L^{(\epsilon,h)} = 0). \tag{3.6}$$

Choose $h = h(\epsilon) = 1/\sqrt{\epsilon}$. Then $\tilde{\rho}_0^{(\epsilon)}(h(\epsilon))$ can be written as $\lambda_0 \int_0^{\sqrt{\epsilon}} y \mathrm{d}B_0(y) + \lambda_0\sqrt{\epsilon}\mathbb{P}(B_0 \geq \sqrt{\epsilon})$, so it can easily be seen that $\tilde{\rho}_0^{(\epsilon)}(h(\epsilon)) \to 0$. Moreover, $\tilde{\rho}_i(h(\epsilon)) = \rho_i(\frac{1}{\sqrt{\epsilon}}) + \lambda_i \frac{1}{\sqrt{\epsilon}}\mathbb{P}(B_i \geq \frac{1}{\sqrt{\epsilon}})$. It can be shown that the last term goes to 0 as $\epsilon \downarrow 0$, so it follows that $\lim_{\epsilon\downarrow 0} \tilde{\rho}_i(h(\epsilon)) = \rho_i$. Thus, if we show that

$$\mathbb{P}(N_1^{(\epsilon,h)} = N_2^{(\epsilon,h)} = \cdots = N_L^{(\epsilon,h)} = 0) \leq \prod_{i=1}^{L}(1 - \tilde{\rho}_i(h)) \text{ for all } \epsilon > 0 \text{ and } h > 0, \tag{3.7}$$

by (3.6) we have that the inequality $\rho_0 - \tilde{\rho}_0^{(\epsilon)}(h(\epsilon)) \leq \prod_{i=1}^{L}(1 - \tilde{\rho}_i(h))$ holds. Letting $\epsilon \downarrow 0$ then gives the desired result $\rho_0 \leq \Pi_{i=1}^{L}(1 - \rho_i)$.

Hence it remains to be shown that (3.7) holds. We do this by comparing the workloads of classes $i = 1, \ldots, L$ in the original system with those in the same system but without class 0. Such a system with only class-$1, \ldots, L$ users will be referred to as the reference system. Since ϵ will remain fixed in the remainder of the proof, we suppress the dependence on ϵ for notational convenience. Let us denote the workload of class i in the h-truncated system at time t by $W_i^h(t)$, and the workload of class i in the h-truncated reference system by $\hat{W}_i^h(t)$. We further represent – both for the original and the reference system – the amount of traffic of class $i = 1, \ldots, L$ that arrives in the time interval (s, t) in the h-truncated system by $A_i^h(s, t)$. For the original system we introduce two more notations. Define $B_0^h(s, t)$ as the amount of service given to class-0 users in (s, t) in the h-truncated system and define $U_i^h(s, t)$ as the capacity wasted in (s, t) in the h-truncated system at node i while there is at least one class-i user that has received at most h in service.
Assume both systems are empty at time $t = 0$. In [6] it is proved that $\hat{W}_i^h(t)$ can be represented as $\hat{W}_i^h(t) = \sup_{s\in[0,t]}\{A_i^h(s, t) - (t - s)\}$. In the reference system the term $(t - s)$ is the total capacity available in the interval (s, t) for work of class i in the h-truncated system. In the original system this capacity is less, since there is also capacity wasted and capacity given to class-0 users with attained service less than h. Hence $W_i^h(t)$ can be represented as

$$W_i^h(t) = \sup_{s\in[0,t]} \{A_i^h(s, t) - [(t - s) - U_i^h(s, t) - B_0^h(s, t)]\}.$$

We can now conclude that for $i = 1, \ldots, L$,

$$W_i^h(t) = \sup_{s\in[0,t]} \{A_i^h(s, t) + U_i^h(s, t) + B_0^h(s, t) - (t - s)\} \geq \sup_{s\in[0,t]} \{A_i^h(s, t) - (t - s)\} = \hat{W}_i^h(t),$$

so that

$$\begin{aligned} \mathbb{P}(N_1^h = \cdots = N_L^h = 0) &= \lim_{t\to\infty} \mathbb{P}(W_1^h(t) = \cdots = W_L^h(t) = 0) \\ &\leq \lim_{t\to\infty} \mathbb{P}(\hat{W}_1^h(t) = \cdots = \hat{W}_L^h(t) = 0) \\ &= \prod_{i=1}^{L}(1 - \tilde{\rho}_i(h)), \end{aligned}$$

where the last equality follows from the independence of the various classes in the reference system. We can conclude that (3.7) holds, which completes the proof. □

For weak LAS, we can even prove that in the limiting regime considered here, $\rho_0 < \Pi_{i=1}^{L}(1 - \rho_i)$ is also a sufficient condition for stability of the ϵ-system.

Proposition 3.3.5 *Assume the service discipline is weak LAS. If* $\rho_0 < \Pi_{i=1}^{L}(1-\rho_i)$*, then there exists an* $\bar{\epsilon}$ *such that the* ϵ*-system is stable, for all* $0 < \epsilon < \bar{\epsilon}$*.*

Proof We will use the same notation as in the proof of Proposition 3.3.4. $N_i^h > 0$ implies that $1 - s_0^h(u) - s_i^h(u) = 0$, since when the service discipline is weak LAS, the unused capacity is re-allocated to class-i users with attained service less than h. $U_i^h(s,t)$ can be written as $\int_s^t (1 - s_0^h(u) - s_i^h(u))\mathbf{1}_{(N_i^h(u)>0)}du$, so for weak LAS, $U_i^h(s,t)$ is equal to 0.
Again focus on the ϵ-system. We can conclude that

$$\begin{aligned} W_i^{h(\epsilon)}(t) &= \sup_{s\in[0,t]} \{A_i^{h(\epsilon)}(s,t) + B_0^{h(\epsilon)}(s,t) - (t-s)\} \\ &\leq \sup_{s\in[0,t]} \{A_i^{h(\epsilon)}(s,t) - (1-g(\epsilon))(t-s)\} \\ &\quad + \sup_{s\in[0,t]} \{B_0^{h(\epsilon)}(s,t) - g(\epsilon)(t-s)\}. \end{aligned} \tag{3.8}$$

Like in the proof of Proposition 3.3.4, choose $h(\epsilon) = \frac{1}{\sqrt{\epsilon}}$, so that $\tilde{\rho}_0^{(\epsilon)}(h(\epsilon)) \to 0$ and $\tilde{\rho}_i(h(\epsilon)) \to \rho_i$ as $\epsilon \downarrow 0$. In addition, let $g(\epsilon)$ be such that $\lim_{\epsilon\downarrow 0} g(\epsilon) = 0$ and $\lim_{\epsilon\downarrow 0} \tilde{\rho}_0^{(\epsilon)}(h(\epsilon))/g(\epsilon) = 0$. By (3.8) we have

$$\begin{aligned} &\mathbb{P}(N_1^{h(\epsilon)}(t) = \ldots = N_L^{h(\epsilon)}(t) = 0) \\ &= \mathbb{P}(W_1^{h(\epsilon)}(t) = \ldots = W_L^{h(\epsilon)}(t) = 0) \\ &\geq \mathbb{P}\Big(\sup_{s\in[0,t]} \{A_i^{h(\epsilon)}(s,t) - (1-g(\epsilon))(t-s)\} = 0,\ \forall i = 1,\ldots,L; \\ &\qquad \sup_{s\in[0,t]} \{B_0^{h(\epsilon)}(s,t) - g(\epsilon)(t-s)\} = 0\Big) \\ &\geq \mathbb{P}\Big(\sup_{s\in[0,t]} \{A_i^{h(\epsilon)}(s,t) - (1-g(\epsilon))(t-s)\} = 0,\ \forall i = 1,\ldots,L\Big) \\ &\quad -\mathbb{P}(\sup_{s\in[0,t]} \{B_0^{h(\epsilon)}(s,t) - g(\epsilon)(t-s)\} > 0). \end{aligned} \tag{3.9}$$

For ϵ small enough, the limit of (3.9) in the time-average sense, as $t \to \infty$, is $\Pi_{i=1}^{L}(1 - \frac{\tilde{\rho}_i(h(\epsilon))}{1-g(\epsilon)})$, using the independence of the arrival processes. Interpreting $\sup_{s\in[0,t]}\{B_0^{h(\epsilon)}(s,t) - g(\epsilon)(t-s)\}$ as the workload in a queue with input process $B_0^{h(\epsilon)}(s,t)$ and constant service rate $g(\epsilon)$, we have that the time-average limit of $\mathbb{P}(\sup_{s\in[0,t]}\{B_0^{h(\epsilon)}(s,t) - g(\epsilon)(t-s)\} > 0)$ can not be more than $\limsup_{t\to\infty} \frac{B_0^{h(\epsilon)}(0,t)}{g(\epsilon)t} \leq \lim_{t\to\infty} \frac{A_0^{h(\epsilon)}(0,t)}{g(\epsilon)t} = \frac{\tilde{\rho}_0^{(\epsilon)}(h(\epsilon))}{g(\epsilon)}$, where the latter limit holds with probability 1. Therefore

$$\mathbb{P}(N_1^{h(\epsilon)} = \ldots = N_L^{h(\epsilon)} = 0) \geq \Pi_{i=1}^{L}\Big(1 - \frac{\tilde{\rho}_i(h(\epsilon))}{1-g(\epsilon)}\Big) - \frac{\tilde{\rho}_0^{(\epsilon)}(h(\epsilon))}{g(\epsilon)}.$$

By choice of $h(\epsilon)$ and $g(\epsilon)$, the quantity on the right-hand side tends to $\Pi_{i=1}^{L}(1-\rho_i)$ as $\epsilon \downarrow 0$. The capacity available in the ϵ-system to serve class-i users with attained service greater than $h(\epsilon)$, $i = 1,\ldots,L$, and to serve class-0 users, is not less than $\mathbb{P}(N_1^{h(\epsilon)} = \ldots = N_L^{h(\epsilon)} = 0)$. By Property 3.0.2 it is therefore sufficient for stability to have

$$\rho_0 + \textstyle\sum_{i=1}^{L}(\rho_i - \tilde{\rho}_i(h(\epsilon))) < \mathbb{P}(N_1^{h(\epsilon)} = \ldots = N_L^{h(\epsilon)} = 0), \tag{3.10}$$

where the term on the left hand side is the traffic load belonging to class-i users with attained service greater than $h(\epsilon)$, $i = 1, \ldots, L$, and to class-0 users. Since $\lim_{\epsilon \downarrow 0} \mathbb{P}(N_1^{h(\epsilon)} = \ldots = N_L^{h(\epsilon)} = 0) \geq \Pi_{i=1}^L (1 - \rho_i)$ and $\lim_{\epsilon \downarrow 0} \tilde{\rho}_i(h(\epsilon)) = \rho_i$, inequality (3.10) holds for ϵ small enough, if we assume that $\rho_0 < \Pi_{i=1}^L (1 - \rho_i)$. This concludes the proof. □

3.3.2 Small class-0 users

In this subsection we consider class-0 users with small service requirements compared to the service requirements of class-i users, $i = 1, \ldots, L$. In Subsection 3.3.1 we derived the necessary stability conditions in a limiting regime where class-0 users become extremely large. We found as the stability condition $\rho_0 < \Pi_{i=1}^L (1 - \rho_i)$, which is more stringent than the standard conditions. In contrast, in this subsection we show that the standard conditions can in fact already be sufficient in case of extremely small class-0 users.
Again, we study a sequence of systems indexed by ϵ and let $\epsilon \downarrow 0$. In this ϵ-system class-0 users arrive according to a Poisson process of rate $\lambda_0^{(\epsilon)} := \lambda_0/\epsilon$ and their sizes are distributed as $B_0^{(\epsilon)} := \epsilon B_0$, that is the class-0 users become extremely small as $\epsilon \downarrow 0$, but the traffic load of class 0, ρ_0, remains constant. The next proposition shows that the standard conditions can be arbitrarily close to being sufficient. We will consider the ϵ-system with $B_0^{(\epsilon)}$ truncated at $h(\epsilon)$, with $\lim_{\epsilon \downarrow 0} h(\epsilon) = 0$.

Proposition 3.3.6 *Suppose the service discipline is weak or strict LAS and assume* $\lim_{\epsilon \downarrow 0} h(\epsilon) = 0$. *If for some* $i = 1, \ldots, L$, $\rho_0 + \rho_i < 1$, *then there exists an* $\bar{\epsilon}$ *such that for all* $0 < \epsilon < \bar{\epsilon}$, *node* i *is stable in the* ϵ*-system with* $B_0^{(\epsilon)}$ *truncated at* $h(\epsilon)$.
Hence, if $\rho_0 + \rho_i < 1$ *for all* $i = 1, \ldots, L$, *then there exists an* $\tilde{\epsilon}$ *such that the* ϵ*-system with* $B_0^{(\epsilon)}$ *truncated at* $h(\epsilon)$ *is stable for all* $0 < \epsilon < \tilde{\epsilon}$.

Proof Assume $\rho_0 + \rho_i < 1$ for an $i = 1, \ldots, L$. We have $\lim_{\epsilon \downarrow 0} h(\epsilon) = 0$, so there is an $\bar{\epsilon}$ such that

$$\rho_0 + \rho_i + \sum_{j=1, j \neq i}^{L} \tilde{\rho}_j(h(\epsilon)) < 1, \text{ for all } \epsilon < \bar{\epsilon}. \tag{3.11}$$

From this it follows that $\tilde{\rho}_0^{(\epsilon)}(h(\epsilon)) + \sum_{j=1}^L \tilde{\rho}_j(h(\epsilon)) < 1$, for all $\epsilon < \bar{\epsilon}$, which, according to Property 3.0.2, is a sufficient condition for stability of the $h(\epsilon)$-truncated system. The service requirements of class 0 in the ϵ-system are bounded by $h(\epsilon)$. According to Property 3.3.2, Q_0 is stable, since Q_0 is stable in the $h(\epsilon)$-truncated system.
For class i, Property 3.3.2 implies that Q_i is stable in the $(h(\epsilon), \infty, \ldots, \infty)$-truncated system if and only if Q_i is stable in the $(h(\epsilon), \ldots, h(\epsilon), \infty, h(\epsilon), \ldots, h(\epsilon))$-truncated system, with ∞ the i-th component. Because of Property 3.0.2 (i), for the latter it is sufficient to have (3.11), which holds for $\epsilon < \bar{\epsilon}$. □

Remark 3.3.7 The fact that $h(\epsilon)$ could be chosen such that $\lim_{\epsilon \downarrow 0} h(\epsilon) = 0$ *and* $\lim_{\epsilon \downarrow 0} h(\epsilon)/\epsilon = \infty$, and thus $\mathbb{P}(B_0^{(\epsilon)} \leq h(\epsilon)) \to 1$, as $\epsilon \downarrow 0$, suggests that the non-truncated ϵ-system can be arbitrarily closely approximated by the truncated one. However, the proof of Proposition 3.3.6 relies on the truncation of $B_0^{(\epsilon)}$. In the particular case that B_0 is bounded from above by a constant M, Proposition 3.3.6 does imply that the condition $\rho_0 + \rho_i < 1$ is sufficient for stability of node i in the ϵ-system for ϵ small enough (take $h(\epsilon) = \epsilon M$).

In the next proposition we compute the limit of $x_i^*(\epsilon)$, for $i = 1, \ldots, L$, in the ϵ-system as $\epsilon \downarrow 0$, in the case that the distribution of B_0 has bounded support. Note that Proposition 3.3.3 does not apply because the distribution of B_0 does not have infinite support. That is why we do not have $x_i^* = x_j^*$.

Proposition 3.3.8 *Let Assumption 3.3.1 be satisfied, $\rho_0 < 1$ and B_0 be bounded by a constant M. For weak and strict LAS*

$$\lim_{\epsilon \downarrow 0} x_i^*(\epsilon) = x_i^*(0) := \sup\{x_i : \rho_0 + \tilde{\rho}_i(x_i) < 1\}, \quad i = 1, \ldots, L.$$

Proof Class 0 in the ϵ-system does not notice a truncation at a constant level x for ϵ small enough, because $B_0^{(\epsilon)}$ is bounded by ϵM which approaches 0 as $\epsilon \downarrow 0$. So for ϵ small enough, we have $\tilde{\rho}_0^{(\epsilon)}(x) = \rho_0$.
Take an $x_i < x_i^*(0)$, so $\rho_0 + \tilde{\rho}_i(x_i) < 1$. From Remark 3.3.7 it follows that there is an $\bar{\epsilon}$ such that for $\epsilon < \bar{\epsilon}$, Q_i is stable in the x_i-truncated system, or equivalently $x_i \leq x_i^*(\epsilon)$ (by definition of $x_i^*(\epsilon)$). So for every $x_i < x_i^*(0)$, there is an $\bar{\epsilon}$ such that $x_i \leq x_i^*(\epsilon)$ for all $\epsilon < \bar{\epsilon}$. We also have $x_i^*(\epsilon) \leq x_i^*(0)$. Together, these two observations imply that $\lim_{\epsilon \downarrow 0} x_i^*(\epsilon) = x_i^*(0)$. □

3.4 Concluding remarks

In this chapter the stability conditions of straightforward extensions of SERPT, SRPT and LAS have been considered in the linear resource-sharing network. We explicitly investigated situations in which the class-0 users were either small or large compared to the users of classes $1, \ldots, L$. Under SERPT, SRPT as well as LAS, we observed that the conditions for stability were the standard ones in case of small class-0 users. However, in case of large class-0 users the stability conditions for the network are $\rho_0 < \Pi_{i=1}^{L}(1 - \rho_i)$, which can be much stricter than the standard ones. This implies that these disciplines are persistently non-work conserving, i.e. there is persistently capacity wasted at the nodes.

In a real resource-sharing network this means that the available resources are used inefficiently, since capacity can be left underutilized even when congestion builds up. This happens as the result of the size-based priorities. When flows with long routes have large service requirements compared to the ones with short routes, instability may already arise for arbitrarily low traffic loads. We can therefore conclude that, in contrast to single-server systems, applying straightforward extensions of size-based service disciplines in resource-sharing networks will not give optimal performance, and may in fact unnecessarily cause network instability.

Chapter 4

Optimal scheduling

In this chapter we consider the linear resource-sharing network with L nodes of Figure 2.1 and seek policies that in some sense appropriate minimize the total number of users in the network. We only allow (possibly preemptive) policies that have no knowledge available of the remaining service requirements. We denote this class of policies by Π. This chapter focuses in particular on the following two policies, which in many situations optimize the performance.

- Policy π^*: This policy serves all classes $i = 1, \ldots, L$, whenever at least one user of each of those classes is present. Otherwise class 0 is served. When Q_0 is empty, classes $i = 1, \ldots, L$ with at least one user present are served.

- Policy π^{**}: This policy gives preemptive priority to class 0. When Q_0 is empty, classes $i = 1, \ldots, L$ with at least one user present are served.

In the remainder of this chapter, π^* and π^{**} will always refer to these policies. Note that under both π^* and π^{**} every node with users contending for service, uses its total capacity. This implies that the average wasted service rate in node i, w_i, equals 0 for all $i = 1, \ldots, L$. In Chapter 3 we derived that $\rho_0 + \rho_i + w_i < 1$ is a sufficient condition for stability of node i. Since $w_i = 0$ for all $i = 1, \ldots, L$, we can conclude that for both policy π^* and π^{**} the system is stable under the standard conditions.

In Section 4.1 some useful definitions and preliminaries are presented. In Section 4.2 we investigate the workload in the network. For general service requirements we obtain sample-path inequalities for the workload under various policies. In Section 4.3 we assume exponential service requirements. In that case, under certain conditions, the workload inequalities allow us to straightforwardly identify a policy that minimizes the mean number of users present in the system at each point in time. In order for this result to hold, there has to be an ordering between the service rates of the class-0 users and the class-$1, 2, \ldots, L$ users. In Section 4.4 we prove a stronger result under the same assumptions as in Section 4.3. Specifically, we use dynamic programming to show that the policies that are optimal in terms of mean queue lengths, are in fact stochastically optimal. In Section 4.5 we generalize some of the results from Section 4.4 to phase-type service requirement distributions.

4.1 Definitions and preliminaries

4.1.1 Stochastic ordering

We use the following definition of a stochastic order relation between two random variables.

Definition 4.1.1 (Stochastic order relation) *A random variable X is stochastically smaller than a random variable Y, notation $X \leq_{st} Y$, if $\mathbb{P}(X > s) \leq \mathbb{P}(Y > s)$ for all s.*

Since $\mathbb{E}(X) = \int \mathbb{P}(X > x)\mathrm{d}x \leq \int \mathbb{P}(Y > x)\mathrm{d}x = \mathbb{E}(Y)$, it follows easily from the definition that $X \leq_{st} Y$ implies $\mathbb{E}(X) \leq \mathbb{E}(Y)$. The following lemma gives a characterization of the stochastic order relation. The proof can be found in [10].

Lemma 4.1.2 *Two random variables X and Y satisfy $X \leq_{st} Y$ if and only if there exist two random variables X' and Y', defined on the same probability space, such that $\mathbb{P}(X' \leq Y') = 1$, $X' \stackrel{d}{=} X$ and $Y' \stackrel{d}{=} Y$. ($X_1 \stackrel{d}{=} X_2$ means $\mathbb{P}(X_1 \leq x) = \mathbb{P}(X_2 \leq x)$ for all x.)*

4.1.2 Phase-type distribution

In a Markov process each state is visited for an exponentially distributed amount of time, after which a transition occurs to one of the other states according to a certain probability distribution. A state is absorbing if the process cannot leave this state once it has entered this state.

Definition 4.1.3 (Phase-type distribution) *A random variable of phase-type is distributed as the time until absorption in a Markov process, starting from the states with a certain initial distribution.*

The set of phase-type distributions is dense in the set of probability distributions on $(0, \infty)$. (See for example [11].) Hence every positive random variable B with distribution function $F(s)$, can be approximated by phase-type distributed random variables B_n with distribution function $F_n(s)$, in the sense of weak convergence, i.e. $\lim_{n\to\infty} F_n(s) = F(s)$ for every s.

4.1.3 Increasing and decreasing failure rates

Let B be a positive random variable with distribution function $F(s)$ and a continuous density function $f(s)$. We can think of B as the service requirement of a user. The probability that the user leaves the system in the next Δs time units, given that it has received an amount of s in service, can then be written as

$$\mathbb{P}(s < B \leq s + \Delta s | B > s) = \frac{\mathbb{P}(s < B \leq s + \Delta s)}{\mathbb{P}(B > s)} = \frac{f(s)\Delta s}{1 - F(s)} + o(\Delta s), \quad \text{as } \Delta s \to 0.$$

In the literature, the failure rate function is defined by

$$r(s) = \frac{f(s)}{1 - F(s)}, \quad \text{for } s \text{ such that } F(s) < 1.$$

Hence, for Δs small enough, $r(s)\Delta s$ is approximately the probability that within the next Δs units of service, the user with attained service s leaves the system.

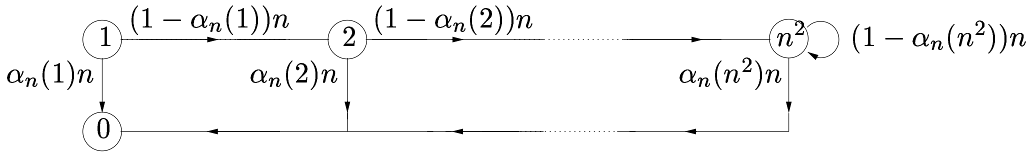

Figure 4.1: The state transition diagram of the Markov process with absorbing state 0.

A random variable is said to have a decreasing failure rate (DFR) if the failure rate function is non-increasing in s. A random variable is said to have a increasing failure rate (IFR) if the failure rate function is non-decreasing in s. When we think of B as the service requirement of a user, this means that in case of DFR (IFR), the more service it has received, the smaller (larger) the probability becomes that it leaves the system in the next Δs time units. Notice that for an exponentially distributed random variable with mean $1/\mu$, the failure rate function is equal to $r(s) = \mu$. Hence it is both IFR and DFR.

As stated in Subsection 4.1.2, every positive random variable can be approximated arbitrarily close by phase-type distributed random variables. We will now explicitly give such a possible approximation. Define B_n as the phase-type distributed random variable, distributed as the time until absorption in the Markov process depicted in Figure 4.1, when starting in state 1. We define

$$\alpha_n(k) = \begin{cases} \frac{F(\frac{k}{n})-F(\frac{k-1}{n})}{\overline{F}(\frac{k-1}{n})} & \text{if } \overline{F}(\frac{k-1}{n}) > 0 \\ 1 & \text{if } \overline{F}(\frac{k-1}{n}) = 0. \end{cases}$$

We can think of B_n as the service requirement of a user as follows. The service of a user starts in state or phase 1. Each phase takes an exponential time with mean $1/n$. After phase k, with probability $\alpha_n(k)$ the user leaves the system and with probability $1-\alpha_n(k)$ it goes to phase $\min(k+1, n^2)$.

Let us denote the distribution function of B_n by $F_n(s)$. The next lemma states that F_n converges weakly to F, that is B can be approximated arbitrarily close by B_n. Furthermore, if the random variable B is DFR (IFR), then $\alpha_n(k)$ is decreasing (increasing) in k. The proof can be found in Appendix B of [3].

Lemma 4.1.4 *The distribution function F_n, as described above, converges weakly to F. If B is DFR (IFR), then $\alpha_n(k)$ is decreasing (increasing) in k, for every n.*

Note that for k such that $\overline{F}(\frac{k-1}{n}) > 0$ we have $\alpha_n(k)n = \frac{\mathbb{P}(\frac{k-1}{n}<B\leq\frac{k}{n})}{\mathbb{P}(B>\frac{k-1}{n})}n = r(\frac{k-1}{n}) + o(1)$ as $n \to \infty$, hence $r(\frac{k-1}{n}) \sim \alpha_n(k)n$ as $n \to \infty$. For service requirements with general distributions, the failure rate can therefore be approximated by the rate at which the user leaves the system when it is in a certain phase of the phase-type distribution B_n, for n large enough.

4.2 Workload

In this section we allow for general service requirement distributions. We are interested in comparing the workloads of the classes under various policies and will derive some inequalities that hold sample-path wise.

Consider a policy, $\bar{\pi}^i$, that is work-conserving in node i, i.e. the capacity of node i is fully used whenever there is a backlog at that node. This policy always minimizes the total workload of users that require service from node i, as is stated in the next observation.

Observation 4.2.1 *Let $\bar{\pi}^i$ be a policy that is work-conserving in node i, (i.e. the capacity of node i is fully used whenever there is a backlog at that node) and let π be an arbitrary policy. Assume $W_0^{\bar{\pi}^i}(0) \leq W_0^{\pi}(0)$, $W_i^{\bar{\pi}^i}(0) \leq W_i^{\pi}(0)$ and that the arrival and service requirement sequences are identical for both policies. Then*

$$W_0^{\bar{\pi}^i}(t) + W_i^{\bar{\pi}^i}(t) \leq W_0^{\pi}(t) + W_i^{\pi}(t), \quad \forall t \geq 0. \tag{4.1}$$

Now consider the policy π^*. Recall that policy π^* serves class i, $i \neq 0$, only if there are no class-0 users present or if at least one user of each of the classes $i = 1, \ldots, L$ is present. Therefore this policy is work conserving in every node, so inequality (4.1) holds for all $i = 1, \ldots, L$. The next lemma states that at each point in time policy π^* also minimizes the aggregate workload in two nodes (these need not always be the same nodes).

Lemma 4.2.2 *Let π be an arbitrary policy. Assume $W_i^{\pi^*}(0) \leq W_i^{\pi}(0)$ for $i = 0, \ldots, L$ and that the arrival and service requirement sequences are identical for both policies. Then at every point in time t, there are classes j and k with $j \neq k$, $j, k = 1, \ldots, L$, such that*

$$W_0^{\pi^*}(t) + W_j^{\pi^*}(t) + W_k^{\pi^*}(t) \leq W_0^{\pi}(t) + W_j^{\pi}(t) + W_k^{\pi}(t). \tag{4.2}$$

Proof We call $W_0^{\pi}(t) + W_j^{\pi}(t) + W_k^{\pi}(t)$ the workload in nodes j and k. The rate at which the workload in nodes j and k is reduced at time t under policy π is denoted by $s_{jk}^{\pi}(t)$. Note that $s_{jk}^{\pi}(t) \in [0, 2]$ and $s_{jk}^{\pi^*}(t) \in \{0, 1, 2\}$. Define t_n and t_n^0 such that $t_0 = t_0^0 = 0$ and

$$t_n = \inf\{t > t_{n-1}^0 : \forall i = 1, \ldots, L \;\; W_i^{\pi^*}(t) > 0\}, \quad t_n^0 = \inf\{t > t_n : \exists i \text{ such that } W_i^{\pi^*}(t) = 0\},$$

for $n = 1, 2, \ldots$. Note that in the interval $[t_n, t_n^0)$ all queues of classes $1, \ldots, L$ are non-empty under policy π^*, while in $[t_n^0, t_{n+1})$ there is at least one such queue empty under policy π^*. The lemma will be proved by induction on t_n. By assumption, the statement holds at time $t = t_0$. Assume the statement holds at time $t < t_n$. We will show that it remains true at time t, $t_n \leq t < t_{n+1}$.
At time t_n^- there exists exactly one $i \in \{1, \ldots, L\}$ such that $W_i^{\pi^*}(t_n^-) = 0$. By induction, there are $j \neq k \in \{1, \ldots, L\}$ such that $W_0^{\pi^*}(t_n^-) + W_j^{\pi^*}(t_n^-) + W_k^{\pi^*}(t_n^-) \leq W_0^{\pi}(t_n^-) + W_j^{\pi}(t_n^-) + W_k^{\pi}(t_n^-)$. At time t_n a class-i user arrives and all queues under policy π^* become non-empty. Hence $s_{jk}^{\pi^*}(t_n) = 2$ and policy π cannot do better. This implies that inequality (4.2) remains true for the workload in nodes j and k, for all $t < t_n^0$. For every $t \in [t_n^0, t_{n+1})$ there is at least one $m = 1, \ldots, L$ such that $W_m^{\pi^*}(t) = 0$. Since π^* is work conserving in all nodes, for an arbitrary node $l \neq m, l \in \{1, \ldots, L\}$, inequality (4.1) holds. This gives for $t_n^0 \leq t < t_{n+1}$

$$W_0^{\pi^*}(t) + W_l^{\pi^*}(t) + W_m^{\pi^*}(t) = W_0^{\pi^*}(t) + W_l^{\pi^*}(t) \leq W_0^{\pi}(t) + W_l^{\pi}(t) \leq W_0^{\pi}(t) + W_l^{\pi}(t) + W_m^{\pi}(t).$$

Therefore inequality (4.2) holds for all $t < t_{n+1}$, which proves the lemma. □

	$W_0^{\pi^1}(t)$	$W_1^{\pi^1}(t)$	$W_2^{\pi^1}(t)$	$W_3^{\pi^1}(t)$	$W_0^{\pi^2}(t)$	$W_1^{\pi^2}(t)$	$W_2^{\pi^2}(t)$	$W_3^{\pi^2}(t)$
$t=0$	10	2	1	0	10	2	1	0
$t=1$	10	1	0	10	9	2	1	10
$t=2$	10	0	0	9	9	1	0	9
$t=3$	9	0	0	9	9	0	0	8

Table 4.1: Example.

For a linear network with two nodes ($L = 2$), the classes j and k in Lemma 4.2.2 are necessarily classes 1 and 2. Therefore we have the following corollary.

Corollary 4.2.3 *In a linear network with $L = 2$ nodes, policy π^* minimizes the total workload in the system sample-path wise.*

Remark 4.2.4 It may seem plausible that for more than two nodes, there also exists a policy that minimizes the total workload in the system sample-path wise. However, this is not the case. In order to minimize the total workload sample-path wise, it is necessary to serve the classes $1, \ldots, L$, whenever there are users present of at least two such classes, since this maximizes the work depletion rate at this moment in time. Nevertheless serving class 0, allows classes $1, \ldots, L$ to be served at an even higher rate later, when new users arrive. Hence it may be better not to operate at the maximum work depletion rate. Therefore no sample-path wise policy that minimizes the total workload can be found.
We illustrate this with the following example for $L = 3$. We compare the policy π^1 that serves classes 1, 2 and 3, whenever there are users present of at least two such classes, with policy π^2 that does not obey that rule. At time $t = 0$, both policies start with the same workload. The arrival and service requirement sequences are identical under both policies. In Table 4.1 the two processes are described up till time $t = 3$. Note that at time $t = 1$ a class-3 user has entered the system with service requirement 10. Policy π^1 maximizes the work depletion rate at every moment in time and has at time $t = 3$ a total workload of 18, while policy π^2 has a total workload of only 17.

We conclude the section with a straightforward observation.

Observation 4.2.5 *Let $\tilde{\pi}^i$ be a policy that gives preemptive priority to class i, for an $i = 1, \ldots, L$ and let π be an arbitrary policy. Assume $W_i^{\tilde{\pi}^i}(0) \leq W_i^{\pi}(0)$ and that the arrival and service requirement sequences are identical for both policies. Then*

$$W_i^{\tilde{\pi}^i}(t) \leq W_i^{\pi}(t), \quad \forall t \geq 0.$$

4.3 Mean queue lengths

In this section we assume the service requirements are exponentially distributed with mean $\beta_i = 1/\mu_i$. In certain cases, depending on the values of the service rates μ_i, $i = 0, \ldots, L$, we can obtain the policy that minimizes the mean number of users in the system at every moment in time, making use of the results for workloads presented in the previous section. Note that such a policy minimizes the mean overall sojourn time as well, because of Little's law.

In a single-server system, the μ-rule is known to be optimal (even stochastically, see [7]). This rule amounts to serving the user with the highest failure rate. For exponential service requirements we have $r_i(s) = \mu_i$ for all s, therefore the μ-rule serves that class with the highest service rate, which makes also intuitively sense.
In the network we discuss, the μ-rule cannot be applied. It is possible that at node i, class i has the highest service rate and at node j, class 0 has the highest service rate. Serving both classes i and 0 at the same time is impossible, since class 0 needs service at nodes i and j simultaneously. This suggests that in the linear network, we should compare the service rate of class 0 with the sum of the service rates of classes $1, \ldots, L$, or equivalently, consider the total output rate of the system.
Besides trying to maximize the total output rate of the system, we need to take into account that class 0 needs simultaneously service of all nodes. Giving priority to certain classes may leave certain resources underutilized. For example, if $\mu_i > \mu_0$ for all $i = 1, \ldots, L$, then giving priority to classes $1, \ldots, L$, always greedily maximizes the total output rate of the system. However in Section 3.1 we saw that such a discipline causes instability when $\Pi_{j=1}^{L}(1 - \rho_j) < \rho_0$, while the service discipline that gives preemptive priority to class 0, can ensure stability in that case. This is a consequence of the non-work conserving behavior of the discipline that gives priority to classes $1, \ldots, L$. When for example only classes 0 and 1 are present and we give priority to class 1, only the capacity at node 1 is fully used. The other nodes are left unutilized, while there is a backlog at these nodes, namely of class 0.
Hence, we conclude that besides trying to maximize the total output rate of the system, another objective of an optimal policy is to achieve the highest possible degree of service parallelism. In general there can be a trade-off between maximizing the output rate and using the full capacity in every node whenever there is a backlog at that node. This makes it difficult to characterize or to find the optimal policy. In Subsections 4.3.1 and 4.3.2 we will prove that for certain values of the service rates an optimal policy can be found, independent of the arrival rates. This concerns exactly the situations in which the above described trade-off does not occur. The results follow rather easily by comparing the workloads under the optimal policy and an arbitrary policy and using Lemma 4.3.1 and results from Section 4.2. In Subsection 4.3.3 we briefly mention the combinations of the service rates not covered in Subsections 4.3.1 and 4.3.2, for which the above described trade-off does occur.

Lemma 4.3.1 *If the service requirements of class i are exponentially distributed with mean $1/\mu_i$, for $i = 0, \ldots, L$ and if for some $I \subseteq \{0, \ldots, L\}$, $\sum_{i \in I} W_i^{\bar{\pi}}(t) \leq \sum_{i \in I} W_i^{\pi}(t)$, $\forall t \geq 0$, and $\bar{\pi}, \pi \in \Pi$, then*

$$\sum_{i \in I} \sum_{k=1}^{N_i^{\bar{\pi}}(t)} \bar{E}_k^i \leq_{st} \sum_{i \in I} \sum_{k=1}^{N_i^{\pi}(t)} E_k^i, \quad \forall\, t \geq 0, \tag{4.3}$$

with $\bar{E}_k^i, E_k^i$ independent identically distributed copies of a random variable $E^i \stackrel{d}{=} Exp(\mu_i)$ and independent of $N_i^{\bar{\pi}}$ and N_i^{π}. Hence, taking expectations

$$\sum_{i \in I} \frac{1}{\mu_i} \mathbb{E}(N_i^{\bar{\pi}}(t)) \leq \sum_{i \in I} \frac{1}{\mu_i} \mathbb{E}(N_i^{\pi}(t)). \tag{4.4}$$

Proof The service requirements of the users are exponentially distributed. Because of the memoryless property and the fact that policies π and $\bar{\pi}$ have no knowledge of the remaining service requirements, the remaining service requirement of a class-i user is exponentially distributed with the same mean $1/\mu_i$. The workload, $W_i(t)$, is therefore distributed as $\sum_{k=1}^{N_i(t)} E_k^i$. Since $\sum_{i\in I} W_i^{\bar{\pi}}(t) \leq \sum_{i\in I} W_i^{\pi}(t)$, the stochastic inequality in (4.3) follows, using Lemma 4.1.2. By Wald's lemma we have $\mathbb{E}(\sum_{k=1}^{N_i(t)} E_k^i) = \frac{1}{\mu_i}\mathbb{E}(N_i(t))$, so (4.4) follows by taking expectations on both sides in (4.3). □

4.3.1 Optimality of policy π^*

In this subsection we consider the situation that $\sum_{i=1}^{L} \mu_i \geq \mu_0$ and $\sum_{i=1,i\neq j}^{L} \mu_i \leq \mu_0$ for $j = 1,\ldots,L$. Recall that policy π^* serves classes $i = 1,\ldots,L$, whenever at least one user of each of these classes is present. Otherwise class 0 is served. When Q_0 is empty, classes $i = 1,\ldots,L$ with at least one user present are served. We see that whenever there is a backlog at a node, its full capacity is used under policy π^*, so π^* is work conserving in all nodes. Furthermore, at every moment in time, the output rate is maximized under policy π^*. So for this choice of service rates no trade-off occurs between maximizing the output rate and being work conserving. This suggests that policy π^* is optimal, as is confirmed by the next proposition.

Proposition 4.3.2 *Assume $W_i^{\pi^*}(0) \leq W_i^{\pi}(0)$, for $i = 0,\ldots,L$. If $\sum_{i=1}^{L} \mu_i \geq \mu_0$ and $\sum_{i=1,i\neq j}^{L} \mu_i \leq \mu_0$ for $j = 1,\ldots,L$, then policy π^* minimizes the mean number of users present at time t, for every $t \geq 0$, i.e. $\mathbb{E}(N^{\pi^*}(t)) \leq \mathbb{E}(N^{\pi}(t))$, $\forall \pi \in \Pi$.*

Proof Consider an arbitrary policy π. Since policy π^* is work conserving in every node, by Observation 4.2.1 we have for $i = 1,\ldots,L$,

$$W_0^{\pi^*}(t) + W_i^{\pi^*}(t) \leq W_0^{\pi}(t) + W_i^{\pi}(t),$$

which gives by Lemma 4.3.1

$$\sum_{k=1}^{N_0^{\pi^*}(t)} \bar{E}_k^0 + \sum_{k=1}^{N_i^{\pi^*}(t)} \bar{E}_k^i \leq_{st} \sum_{k=1}^{N_0^{\pi}(t)} E_k^0 + \sum_{k=1}^{N_i^{\pi}(t)} E_k^i$$

and hence

$$\frac{1}{\mu_0}\mathbb{E}(N_0^{\pi^*}(t)) + \frac{1}{\mu_i}\mathbb{E}(N_i^{\pi^*}(t)) \leq \frac{1}{\mu_0}\mathbb{E}(N_0^{\pi}(t)) + \frac{1}{\mu_i}\mathbb{E}(N_i^{\pi}(t)). \tag{4.5}$$

In the same way, from Lemmas 4.2.2 and 4.3.1 we can conclude that at time t there are classes j and k, $j \neq k \in \{1,\ldots,L\}$, such that

$$\frac{1}{\mu_0}\mathbb{E}(N_0^{\pi^*}(t)) + \frac{1}{\mu_j}\mathbb{E}(N_j^{\pi^*}(t)) + \frac{1}{\mu_k}\mathbb{E}(N_k^{\pi^*}(t)) \leq \frac{1}{\mu_0}\mathbb{E}(N_0^{\pi}(t)) + \frac{1}{\mu_j}\mathbb{E}(N_j^{\pi}(t)) + \frac{1}{\mu_k}\mathbb{E}(N_k^{\pi}(t)). \tag{4.6}$$

When we now multiply inequality (4.5) by $\mu_0 - \sum_{l=1,l\neq i}^{L} \mu_l \geq 0$, for $i = j,k$ and by μ_i for all $i = 1,\ldots,L$ with $i \neq j,k$, multiply inequality (4.6) by $\sum_{i=1}^{L} \mu_i - \mu_0 \geq 0$ and sum these $L+1$ inequalities, this gives $\sum_{i=0}^{L} \mathbb{E}(N_i^{\pi^*}(t)) \leq \sum_{i=0}^{L} \mathbb{E}(N_i^{\pi}(t))$. □

Remark 4.3.3 In the previous proposition we have used a stochastic inequality to prove an inequality in expectation for $N(t)$. It might seem possible to readily derive a stochastic inequality for $N(t)$ as well. However this is not that clear. For example take $L = 2$ and $\mu_i = \mu$, for $i = 0, 1, 2$. Lemma 4.2.2 implies that $\sum_{i=0}^{2} W_i^{\pi^*}(t) \leq \sum_{i=0}^{2} W_i^{\pi}(t)$ and Lemma 4.3.1 then gives

$$\sum_{k=1}^{N^{\pi^*}(t)} \bar{E}_k \leq_{st} \sum_{k=1}^{N^{\pi}(t)} E_k, \quad \forall\, t > 0,$$

with $\bar{E}_k, E_k$ independent identically distributed copies of a random variable $E \stackrel{d}{=} Exp(\mu)$. One may be tempted to conclude that $N^{\pi^*}(t) \leq_{st} N^{\pi}(t)$, for all $t > 0$. However in general, the inequality $\sum_{k=1}^{K^*} \bar{E}_k \leq_{st} \sum_{k=1}^{K} E_k$ is not sufficient to imply $K^* \leq_{st} K$ as illustrated by the following counterexample. We construct two random variables K^* and K that are not stochastically ordered, while $\sum_{k=1}^{K^*} \bar{E}_k \leq_{st} \sum_{k=1}^{K} E_k$ is satisfied. Take

$$K^* = \begin{cases} 0 & \text{with probability} \quad 1 - 2\epsilon \\ 10 & \text{with probability} \quad \epsilon \\ 11 & \text{with probability} \quad \epsilon \end{cases}$$

and

$$K = \begin{cases} 10 & \text{with probability} \quad 1 - \frac{3}{2}\epsilon \\ 11 & \text{with probability} \quad \frac{3}{2}\epsilon, \end{cases}$$

with $\epsilon > 0$. It can be checked that there exists no stochastic ordering between K^* and K by taking both $s = 9$ and $s = 10$ in Definition 4.1.1. In order to verify that $\sum_{k=1}^{K^*} \bar{E}_k \leq_{st} \sum_{k=1}^{K} E_k$, observe that the sum of n exponential distributed random variables with parameter μ, is Erlang distributed with parameters n and μ, so

$$\begin{aligned} \lim_{x\to\infty} \frac{\mathbb{P}(\sum_{k=1}^{K} E_k > x)}{\mathbb{P}(\sum_{k=1}^{K^*} \bar{E}_k > x)} &= \lim_{x\to\infty} \frac{(1 - \frac{3}{2}\epsilon) \sum_{k=1}^{9} e^{-\mu x} \frac{(\mu x)^k}{k!} + \frac{3}{2}\epsilon \sum_{k=1}^{10} e^{-\mu x} \frac{(\mu x)^k}{k!}}{\epsilon \sum_{k=1}^{9} e^{-\mu x} \frac{(\mu x)^k}{k!} + \epsilon \sum_{k=1}^{10} e^{-\mu x} \frac{(\mu x)^k}{k!}} \\ &= \frac{3}{2}. \end{aligned}$$

Hence, there exists an $\bar{x} > 0$ such that $\mathbb{P}(\sum_{k=1}^{K^*} \bar{E}_k > x) \leq \mathbb{P}(\sum_{k=1}^{K} E_k > x)$, for all $x > \bar{x}$. Now choose $\epsilon = \frac{1}{2} \inf_{x \leq \bar{x}} \frac{\mathbb{P}(\sum_{k=1}^{10} E_k > x)}{\mathbb{P}(\sum_{k=1}^{11} E_k > x)} > 0$. We then have for $x \leq \bar{x}$

$$\begin{aligned} \mathbb{P}(\sum_{k=1}^{K^*} \bar{E}_k > x) &= \epsilon \mathbb{P}(\sum_{k=1}^{10} E_k > x) + \epsilon \mathbb{P}(\sum_{k=1}^{11} E_k > x) \\ &\leq 2\epsilon \mathbb{P}(\sum_{k=1}^{11} E_k > x) \leq \mathbb{P}(\sum_{k=1}^{10} E_k > x) \\ &\leq \mathbb{P}(\sum_{k=1}^{K} E_k > x). \end{aligned}$$

This implies $\sum_{k=1}^{K^*} \bar{E}_k \leq_{st} \sum_{k=1}^{K} E_k$, which concludes our counterexample.

4.3.2 Optimality of policy π^{**}

In this subsection we consider the situation $\sum_{i=1}^{L} \mu_i \leq \mu_0$. The policy π^{**} gives preemptive priority to class 0 and is therefore work conserving in all nodes. At the same time, it maximizes the output rate in every node, at every moment in time. It is natural to expect that for this choice of service rates, policy π^{**} minimizes the mean number of users in the system at every moment in time. In the next proposition we prove that this is indeed the case.

Proposition 4.3.4 *Assume $W_i^{\pi^{**}}(0) \leq W_i^{\pi}(0)$, for $i = 0, \ldots, L$. If $\sum_{i=1}^{L} \mu_i \leq \mu_0$, then policy π^{**} minimizes the mean number of users present at time t, for every $t \geq 0$, i.e. $\mathbb{E}(N^{\pi^{**}}(t)) \leq \mathbb{E}(N^{\pi}(t))$, $\forall \pi \in \Pi$.*

Proof We will prove this in the same way as in the previous subsection. Consider an arbitrary policy π. Since policy π^{**} is work conserving in every node, it follows from Observation 4.2.1 and Lemma 4.3.1 that

$$\frac{1}{\mu_0}\mathbb{E}(N_0^{\pi^{**}}(t)) + \frac{1}{\mu_i}\mathbb{E}(N_i^{\pi^{**}}(t)) \leq \frac{1}{\mu_0}\mathbb{E}(N_0^{\pi}(t)) + \frac{1}{\mu_i}\mathbb{E}(N_i^{\pi}(t)), \quad \forall\, i = 1, \ldots, L. \tag{4.7}$$

Observation 4.2.5 and Lemma 4.3.1 imply

$$\mathbb{E}(N_0^{\pi^{**}}(t)) \leq \mathbb{E}(N_0^{\pi}(t)). \tag{4.8}$$

Multiplying (4.7) by $\mu_i \geq 0$, for all $i = 1, \ldots, L$, multiplying (4.8) by $\frac{\mu_0 - \sum_{i=1}^{L} \mu_i}{\mu_0} \geq 0$ and summing these $L + 1$ inequalities gives $\sum_{i=0}^{L} \mathbb{E}(N_i^{\pi^{**}}(t)) \leq \sum_{i=0}^{L} \mathbb{E}(N_i^{\pi}(t))$. □

4.3.3 Discussion of other cases

In the previous subsections optimal policies could be determined independent of the arrival rates λ_i, $i = 0, \ldots, L$. This was only possible for certain choices of the service rates, namely exactly those μ_i's for which $\sum_{i=1, i \neq j}^{L} \mu_i \leq \mu_0$ for all $j = 1, \ldots, L$. When this is not the case, there is no obvious optimal policy and it will probably no longer be independent of λ_i, $i = 0, \ldots, L$. Suppose for example that there is exactly one j such that $\sum_{i=1, i \neq j}^{L} \mu_i \geq \mu_0$. When there are users of all classes but j present, there is no obvious optimal action. Maximizing the total output rate of the system would suggest to serve classes $i = 1 \ldots, L$, $i \neq j$, while being work conserving would suggest to serve class 0. Therefore an optimal policy will probably not only depend on the service rates, but on the arrival rates as well. In the case that λ_j is relatively small, it will be better to serve classes $i = 1, \ldots, L$, $i \neq j$, since waiting for this new class-j user takes too long. In the case that λ_j is relatively large, it will be advantageous to wait for the class-j user and first serve class 0, until a class-j user arrives. In Chapter 5 this situation will be further investigated.

4.4 Stochastic optimality for exponential service requirements

In this section we again assume exponential service requirements. In the previous section we saw that when $\sum_{i=1, i \neq j}^{L} \mu_i \leq \mu_0$ for all $j = 1, \ldots, L$, there is a policy, either π^* or π^{**}, that is optimal in expectation. This result can be strengthened. It can be shown that π^* and π^{**} are stochastically optimal as well. For convenience we only present the results and proofs for the case $L = 2$.

Definition 4.4.1 (Stochastic optimality of a policy) *Policy $\overline{\pi}$ is called stochastically optimal if $N^{\overline{\pi}}(t) \leq_{st} N^{\pi}(t)$, for all $t \geq 0$ and for all policies $\pi \in \Pi$, when $\mathbf{N}^{\overline{\pi}}(0) = \mathbf{N}^{\pi}(0)$. By definition of $\leq_{st}$ this means that $\mathbb{P}(N^{\overline{\pi}}(t) > s | \mathbf{N}^{\overline{\pi}}(0) = \mathbf{n}_0) \leq \mathbb{P}(N^{\pi}(t) > s | \mathbf{N}^{\pi}(0) = \mathbf{n}_0)$, for all $s > 0$, for all $t > 0$ and $\mathbf{n}_0 \in \mathbb{R}_+^{L+1}$.*

In order to investigate stochastic optimality, we focus on the uniformized chain. That is, transition epochs (possibly self-transitions) are generated by a Poisson process of uniform rate $\nu = \sum_{i=0}^{2} \lambda_i + \sum_{i=0}^{2} \mu_i$. Let $\{\mathbf{N}_k\}$ be the corresponding embedded discrete-time Markov chain. Uniformization implies that $\mathbb{P}(\mathbf{N}(t) = \mathbf{n} | \mathbf{N}(0) = \mathbf{n}_0) = \mathbb{P}(\hat{\mathbf{N}}(t) = \mathbf{n} | \hat{\mathbf{N}}(0) = \mathbf{n}_0)$, with $\hat{\mathbf{N}}(t) = \mathbf{N}_{\Lambda(t)}$ and $\{\Lambda(t), t \geq 0\}$ a Poisson process with rate ν. So we have

$$\mathbb{P}(N^{\pi}(t) > s | \mathbf{N}^{\pi}(0) = \mathbf{n}_0) = \sum_{m=0}^{\infty} \mathbb{P}(\Lambda(t) = m) \mathbb{P}(N_m^{\pi} > s | \mathbf{N}_0^{\pi} = \mathbf{n}_0). \tag{4.9}$$

We are interested in a stochastically optimal policy $\overline{\pi}$. By definition, $\overline{\pi}$ should satisfy

$$\mathbb{P}(N^{\overline{\pi}}(t) > s | \mathbf{N}^{\overline{\pi}}(0) = \mathbf{n}_0) \leq \mathbb{P}(N^{\pi}(t) > s | \mathbf{N}^{\pi}(0) = \mathbf{n}_0), \text{ for all } t \geq 0, \quad s > 0, \quad \pi \in \Pi. \tag{4.10}$$

By (4.9), if $\mathbb{P}(N_m^{\overline{\pi}} > s | \mathbf{N}_0^{\overline{\pi}} = \mathbf{n}_0) \leq \mathbb{P}(N_m^{\pi} > s | \mathbf{N}_0^{\pi} = \mathbf{n}_0)$, for all $s > 0$ (i.e. $N_m^{\pi^*} \leq_{st} N_m^{\pi}$), for all $m = 1, 2, \ldots$, then (4.10) holds. To prove that policies π^* and π^{**} are stochastically optimal, we therefore only have to prove that these policies are stochastically optimal in the embedded discrete-time Markov chain $\{\mathbf{N}_m\}$. For the latter we will use the dynamic programming technique (DP).

For the discrete-time Markov chain we consider a finite time horizon M with costs at the end of the time horizon, which are equal to $c(i, j, k)$. Here i is the number of class-0 users, j is the number of class-1 users and k is the number of class-2 users. Assume, without loss of generality, that $\nu = 1$. The DP equation can be written as

$$\begin{aligned} V_0(i, j, k) &= c(i, j, k) \\ V_{n+1}(i, j, k) &= \lambda_0 V_n(i+1, j, k) + \lambda_1 V_n(i, j+1, k) + \lambda_2 V_n(i, j, k+1) \\ &\quad + \min\{\mu_0 V_n((i-1)^+, j, k) + \mu_1 V_n(i, j, k) + \mu_2 V_n(i, j, k), \\ &\quad \mu_0 V_n(i, j, k) + \mu_1 V_n(i, (j-1)^+, k) + \mu_2 V_n(i, j, (k-1)^+)\}, \end{aligned} \tag{4.11}$$

for $n = 0, 1, \ldots, M-1$. Here $V_{n+1}(i, j, k)$ represents the minimum achievable expected cost after a time horizon of $n+1$, when at time 0 the chain starts in state (i, j, k). The minimizing action in (4.11) is the optimal action at $n+1$ time units from the horizon, when the objective is to minimize the expected costs after $n+1$ time units.

When we choose $c(i, j, k) = \mathbf{1}_{(i+j+k>s)}$, the costs after M time units when starting at time 0 in state $\mathbf{n}_0 \in \mathbb{R}_+^3$ are equal to $\mathbb{P}(N_M^{\pi} > s | \mathbf{N}_0^{\pi} = \mathbf{n}_0)$. Hence $V_M(\mathbf{n}_0)$ is equal to $\min_{\pi \in \Pi} \mathbb{P}(N_M^{\pi} > s | \mathbf{N}_0^{\pi} = \mathbf{n}_0)$ and the corresponding minimizing policy consists of the actions chosen in the DP equation (4.11). When we can show that this optimal policy is independent of the value s and the time horizon M, then the stochastic optimality is proved. Namely, then we have found a stationary policy that for every value s minimizes $\mathbb{P}(N_m^{\pi} > s | \mathbf{N}_0^{\pi} = \mathbf{n}_0)$ for all m. A policy is stationary if each decision depends only on the current state information and for instance not on the remaining time horizon.

In the next four lemmas we establish several properties of V_n, under certain conditions on $c(i,j,k)$, from which we can obtain optimal policies.

Lemma 4.4.2 *If $c(i,j,k)$ is non-decreasing in i,j and k, then V_n is non-decreasing in i,j and k for all n.*

Proof The statement follows directly from (4.11). □

The next lemma shows that under certain conditions on $c(i,j,k)$ it is better to serve class 0 rather than classes 1 or 2 alone, independent of the remaining time horizon.

Lemma 4.4.3 *If $c(i,j,k)$ is non-decreasing in i,j and k and $W=c$ satisfies*

$$\mu_0 W(i-1,j,k)+\mu_1 W(i,j,k)\leq \mu_0 W(i,j,k)+\mu_1 W(i,j-1,k), \quad \forall i,j>0,k\geq 0, \tag{4.12}$$

then (4.12) with $W=V_n$ holds for all n.
If $c(i,j,k)$ is non-decreasing in i,j and k and $W=c$ satisfies

$$\mu_0 W(i-1,j,k)+\mu_2 W(i,j,k)\leq \mu_0 W(i,j,k)+\mu_2 W(i,j,k-1), \quad \forall i,k>0,j\geq 0, \tag{4.13}$$

then (4.13) with $W=V_n$ holds for all n.

Proof We will only prove (4.12), since the proof of (4.13) is similar. We will do this by induction on n. For $W=V_0$ it holds. Assume it holds for $W=V_n$. We now prove that it also holds for $W=V_{n+1}$. We have

$$\begin{aligned}
&\mu_0 V_{n+1}(i-1,j,k)+\mu_1 V_{n+1}(i,j,k)\\
&\leq \mu_0[\lambda_0 V_n(i,j,k)+\lambda_1 V_n(i-1,j+1,k)+\lambda_2 V_n(i-1,j,k+1)\\
&\quad +\mu_0 V_n(i-1,j,k)+\mu_1 V_n(i-1,j-1,k)+\mu_2 V_n(i-1,j,(k-1)^+)]\\
&\quad +\mu_1[\lambda_0 V_n(i+1,j,k)+\lambda_1 V_n(i,j+1,k)+\lambda_2 V_n(i,j,k+1)\\
&\quad +\mu_0 V_n(i,j,k)+\mu_1 V_n(i,j-1,k)+\mu_2 V_n(i,j,(k-1)^+)]\\
&=\lambda_0[\mu_0 V_n(i,j,k)+\mu_1 V_n(i+1,j,k)]\\
&\quad +\lambda_1[\mu_0 V_n(i-1,j+1,k)+\mu_1 V_n(i,j+1,k)]\\
&\quad +\lambda_2[\mu_0 V_n(i-1,j,k+1)+\mu_1 V_n(i,j,k+1)]\\
&\quad +\mu_0[\mu_0 V_n(i-1,j,k)+\mu_1 V_n(i,j,k)]\\
&\quad +\mu_1[\mu_0 V_n(i-1,j-1,k)+\mu_1 V_n(i,j-1,k)]\\
&\quad +\mu_2[\mu_0 V_n(i-1,j,(k-1)^+)+\mu_1 V_n(i,j,(k-1)^+)]\\
&\leq \lambda_0[\mu_0 V_n(i+1,j,k)+\mu_1 V_n(i+1,j-1,k)]\\
&\quad +\lambda_1[\mu_0 V_n(i,j+1,k)+\mu_1 V_n(i,j,k)]\\
&\quad +\lambda_2[\mu_0 V_n(i,j,k+1)+\mu_1 V_n(i,j-1,k+1)]\\
&\quad +\mu_0[\mu_0 V_n(i-1,j,k)+\mu_1 V_n(i,j,k)]\\
&\quad +\mu_1[\mu_0 V_n(i-1,j-1,k)+\mu_1 V_n(i,j-1,k)]\\
&\quad +\mu_2[\mu_0 V_n(i-1,j,(k-1)^+)+\mu_1 V_n(i,j,(k-1)^+)],
\end{aligned} \tag{4.14}$$

where the last inequality follows from (4.12). For the state (i, j, k) we do not know which action is the minimizer in the DP equation. If serving class 0 is the minimizer, then

$$\mu_0[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) + \mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k)]$$
$$= \mu_0[V_{n+1}(i,j,k) - \mu_2 V_n(i,j,k)].$$

If serving classes 1 and 2 is the minimizer, then by (4.12)

$$\mu_0[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) + \mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k)]$$
$$\leq \mu_0[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) + \mu_0 V_n(i,j,k) + \mu_1 V_n(i,j-1,k)]$$
$$= \mu_0[V_{n+1}(i,j,k) - \mu_2 V_n(i,j,(k-1)^+)].$$

By Lemma 4.4.2, V_n is non-decreasing, so we can conclude that

$$\mu_0[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) + \mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k)]$$
$$\leq \mu_0[V_{n+1}(i,j,k) - \mu_2 V_n(i,j,(k-1)^+)].$$

Similarly for the state $(i, j-1, k)$ we deduce that

$$\mu_1[\lambda_0 V_n(i+1,j-1,k) + \lambda_1 V_n(i,j,k) + \lambda_2 V_n(i,j-1,k+1)$$
$$+\mu_0 V_n(i-1,j-1,k) + \mu_1 V_n(i,j-1,k)]$$
$$\leq \mu_1[V_{n+1}(i,j-1,k) - \mu_2 V_n(i,j-1,(k-1)^+)].$$

Together with (4.14) this gives

$$\begin{aligned}
\mu_0 V_{n+1}(i-1,j,k) + \mu_1 V_{n+1}(i,j,k) \;&\leq\; \mu_0 V_{n+1}(i,j,k) + \mu_1 V_{n+1}(i,j-1,k) \\
&\quad +\mu_2[\mu_0 V_n(i-1,j,(k-1)^+) + \mu_1 V_n(i,j,(k-1)^+) \\
&\qquad -\mu_0 V_n(i,j,(k-1)^+) - \mu_1 V_n(i,j-1,(k-1)^+)] \\
&\leq\; \mu_0 V_{n+1}(i,j,k) + \mu_1 V_{n+1}(i,j-1,k),
\end{aligned}$$

where the last inequality follows from (4.12). This proves that V_{n+1} satisfies (4.12) as well. □

The next lemma shows that under certain conditions on $c(i, j, k)$, the optimal action is to always serve classes 1 and 2 whenever there are users of both classes present, independent of the remaining time horizon.

Lemma 4.4.4 *If $c(i, j, k)$ is non-decreasing in i, j and k and, for all $i, j, k > 0$, $W = c$ satisfies*

$$\mu_0 W(i,j,k) + \mu_1 W(i,j-1,k) + \mu_2 W(i,j,k-1) \leq \mu_0 W(i-1,j,k) + (\mu_1+\mu_2) W(i,j,k), \quad (4.15)$$

then (4.15) with $W = V_n$ holds for all n.

Proof We will prove the lemma by induction. For $W = V_0$ the statement holds. Suppose it holds for $W = V_n$. We have

$$\begin{aligned}
&\mu_0 V_{n+1}(i,j,k) + \mu_1 V_{n+1}(i,j-1,k) + \mu_2 V_{n+1}(i,j,k-1) \\
&\le \mu_0[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) \\
&\quad +\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k)] \\
&\quad +\mu_1[\lambda_0 V_n(i+1,j-1,k) + \lambda_1 V_n(i,j,k) + \lambda_2 V_n(i,j-1,k+1) \\
&\quad +\mu_0 V_n(i-1,j-1,k) + \mu_1 V_n(i,j-1,k) + \mu_2 V_n(i,j-1,k)] \\
&\quad +\mu_2[\lambda_0 V_n(i+1,j,k-1) + \lambda_1 V_n(i,j+1,k-1) + \lambda_2 V_n(i,j,k) \\
&\quad +\mu_0 V_n(i-1,j,k-1) + \mu_1 V_n(i,j,k-1) + \mu_2 V_n(i,j,k-1)] \\
&= \mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i-1,j-1,k) + \mu_2 V_n(i-1,j,k-1)] \\
&\quad +\mu_1[\mu_0 V_n(i,j,k) + \mu_1 V_n(i,j-1,k) + \mu_2 V_n(i,j,k-1)] \\
&\quad +\mu_2[\mu_0 V_n(i,j,k) + \mu_1 V_n(i,j-1,k) + \mu_2 V_n(i,j,k-1)] \\
&\quad +\lambda_0[\mu_0 V_n(i+1,j,k) + \mu_1 V_n(i+1,j-1,k) + \mu_2 V_n(i+1,j,k-1)] \\
&\quad +\lambda_1[\mu_0 V_n(i,j+1,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j+1,k-1)] \\
&\quad +\lambda_2[\mu_0 V_n(i,j,k+1) + \mu_1 V_n(i,j-1,k+1) + \mu_2 V_n(i,j,k)]. \qquad (4.16)
\end{aligned}$$

Since V_n satisfies (4.15), the terms multiplied by λ_0, λ_1 and λ_2 are smaller than or equal to

$$\begin{aligned}
&\lambda_0[\mu_0 V_n(i,j,k) + (\mu_1+\mu_2) V_n(i+1,j,k)] \\
&+\lambda_1[\mu_0 V_n(i-1,j+1,k) + (\mu_1+\mu_2) V_n(i,j+1,k)] \\
&+\lambda_2[\mu_0 V_n(i-1,j,k+1) + (\mu_1+\mu_2) V_n(i,j,k+1)].
\end{aligned}$$

Combining with (4.16) and rearranging the terms gives

$$\begin{aligned}
&\mu_0 V_{n+1}(i,j,k) + \mu_1 V_{n+1}(i,j-1,k) + \mu_2 V_{n+1}(i,j,k-1) \\
&\le \mu_0[\lambda_0 V_n(i,j,k) + \lambda_1 V_n(i-1,j+1,k) + \lambda_2 V_n(i-1,j,k+1) \\
&\quad +\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i-1,j-1,k) + \mu_2 V_n(i-1,j,k-1)] \\
&\quad +\mu_1[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) \\
&\quad +\mu_0 V_n(i,j,k) + \mu_1 V_n(i,j-1,k) + \mu_2 V_n(i,j,k-1)] \\
&\quad +\mu_2[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) \\
&\quad +\mu_0 V_n(i,j,k) + \mu_1 V_n(i,j-1,k) + \mu_2 V_n(i,j,k-1)] \\
&= \mu_0 V_{n+1}(i-1,j,k) + \mu_1 V_{n+1}(i,j,k) + \mu_2 V_{n+1}(i,j,k),
\end{aligned}$$

where in the last step we use that V_n satisfies (4.15) and that V_n is non-decreasing (by Lemma 4.4.2). This implies that $W = V_{n+1}$ satisfies (4.15) as well. □

Combining Lemmas 4.4.2, 4.4.3 and 4.4.4 results in a stochastically optimal policy for certain choices of the service rates.

Corollary 4.4.5 *If $\mu_1, \mu_2 \leq \mu_0$ and $\mu_1 + \mu_2 \geq \mu_0$, then policy π^* is stochastically optimal.*

Proof If $\mu_1, \mu_2 \leq \mu_0$ and $\mu_1 + \mu_2 \geq \mu_0$, then $c(i,j,k) = \mathbf{1}_{(i+j+k>s)}$ satisfies (4.12), (4.13), (4.15) and is non-decreasing in i, j and k. For every time horizon M and every value s we have the following. Lemma 4.4.2 implies that it is always optimal to serve classes 1 and 2, whenever only users of classes 1 and 2 are present. Lemma 4.4.3 implies that it is always optimal to serve class 0 whenever users of class 0 are present and no users of either class 1 or class 2 are present. Lemma 4.4.4 implies that it is always optimal to serve classes 1 and 2, whenever users of all classes are present.
The stationary policy described above corresponds to π^* and is optimal for every value s and every time horizon M. Therefore policy π^* stochastically minimizes the total number of users in the system. □

In the following lemma we show that under certain conditions, it can also be better to serve class 0 rather than classes 1 and 2 together. Again this is independent of the remaining time horizon.

Lemma 4.4.6 *If $c(i,j,k)$ satisfies (4.12), (4.13) and, for all $i,j,k > 0$, $W = c$ satisfies*

$$\mu_0 W(i-1,j,k) + (\mu_1+\mu_2)W(i,j,k) \leq \mu_0 W(i,j,k) + \mu_1 W(i,j-1,k) + \mu_2 W(i,j,k-1), \quad (4.17)$$

then (4.17) with $W = V_n$ holds for all n.

Proof We will prove the lemma by induction on n. For $W = V_0$ the statement holds. Assume it holds for $W = V_n$. We then have

$$\begin{aligned}
&\mu_0 V_{n+1}(i-1,j,k) + (\mu_1+\mu_2)V_{n+1}(i,j,k) \\
&\leq \mu_0[\lambda_0 V_n(i,j,k) + \lambda_1 V_n(i-1,j+1,k) + \lambda_2 V_n(i-1,j,k+1) \\
&\quad +\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i-1,j-1,k) + \mu_2 V_n(i-1,j,k-1)] \\
&\quad +(\mu_1+\mu_2)[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) \\
&\quad +\mu_0 V_n(i,j,k) + \mu_1 V_n(i,j-1,k) + \mu_2 V_n(i,j,k-1)] \\
&\leq \mu_0[\lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) \\
&\quad +\mu_0 V_n(i-1,j,k) + (\mu_1+\mu_2)V_n(i,j,k)] && (4.18) \\
&\quad +\mu_1[\lambda_0 V_n(i+1,j-1,k) + \lambda_1 V_n(i,j,k) + \lambda_2 V_n(i,j-1,k+1) \\
&\quad +\mu_0 V_n(i-1,j-1,k) + (\mu_1+\mu_2)V_n(i,j-1,k)] && (4.19) \\
&\quad +\mu_2[\lambda_0 V_n(i+1,j,k-1) + \lambda_1 V_n(i,j+1,k-1) + \lambda_2 V_n(i,j,k) \\
&\quad +\mu_0 V_n(i-1,j,k-1) + (\mu_1+\mu_2)V_n(i,j,k-1)], && (4.20)
\end{aligned}$$

where the last inequality follows from (4.17) applied to the terms multiplied by λ_0, λ_1 and λ_2.
By (4.17), the expression (4.18) is equal to $\mu_0 V_{n+1}(i,j,k)$. Note that Lemma 4.4.3 implies that V_n satisfies (4.12) and (4.13). If $j > 1$, because of (4.17), the expression (4.19) is equal to $\mu_1 V_{n+1}(i,j-1,k)$ and if $j = 1$, it follows from (4.13) that (4.19) is equal to $\mu_1 V_{n+1}(i,j-1,k)$ as well. Similarly, the expression in (4.20) is equal to $\mu_2 V_{n+1}(i,j,k-1)$, because of (4.12) and (4.17). This implies that $W = V_{n+1}$ satisfies (4.17) as well. □

Lemmas 4.4.2, 4.4.3 and 4.4.6 result in an optimal policy for certain parameter values.

Corollary 4.4.7 *If $\mu_1 + \mu_2 \leq \mu_0$, then policy π^{**} is stochastically optimal.*

Proof If $\mu_1 + \mu_2 \leq \mu_0$, then $c(i,j,k) = \mathbf{1}_{(i+j+k>s)}$ satisfies (4.12), (4.13), (4.17) and is non-decreasing in i, j and k. Lemmas 4.4.2 and 4.4.3 imply that it is optimal to serve class 0 whenever users of class 0 are present and no users of classes 1 or 2 are present. Lemma 4.4.6 implies that it is optimal to serve class 0 whenever users of all classes are present. The policy described above corresponds to π^{**} and is optimal for every value s and every time horizon M. Therefore policy π^{**} stochastically minimizes the total number of users in the system. $\square$

4.5 Stochastic optimality for phase-type service requirements

In this section we assume that the service requirement distributions of every class are of phase-type. A policy has information available on the phase in which each user is. For certain situations we can (partially) characterize which user should be served to possibly achieve stochastic optimality. Again we need to impose restrictions on the service requirement distributions.

We will use the following notation. Phase k_i of a class-i service requirement takes an exponentially distributed amount of time with mean $1/\mu_{i,k_i}$. Denote by $p^i_{k_i,k'_i}$ the probability that a class-i user in phase k_i makes a transition from the k_i-th phase to the k'_i-th phase, when the k_i-th phase is finished. The index $k_i = 0$ indicates that the user arrives into the system and the index $k'_i = 0$ indicates that the user leaves the system. The probabilities satisfy $\sum_{k'_i=0}^{f_i} p^i_{k_i,k'_i} = 1$, with f_i the number of phases in the distribution of a class-i service requirement. Define the failure rate of a class-i user in phase k_i as $\mu_i(k_i) := \mu_{i,k_i} p^i_{k_i,0}$, which represents the rate with which it leaves the system when it is in phase k_i. Class-i users arrive according to a Poisson process of rate λ_i and with probability p^i_{0,k'_i} they start service in the k'_i-th phase.

In this section we consider special phase-type distributions. In the spirit of IFR and DFR distributions, we define the following two types:

- A phase-type distribution is of IFR-type if $\mu_i(k_i)$ is increasing in k_i and the phases can be indexed such that $p^i_{k_i,0} + p^i_{k_i,k_i} + p^i_{k_i,k_i+1} = 1$.
- A phase-type distribution is of DFR-type if $\mu_i(k_i)$ is decreasing in k_i and the phases can be indexed such that $p^i_{k_i,\overline{k}_i} > 0$ if and only if $k_i \leq \overline{k}_i$ or $\overline{k}_i = 0$.

The reason why these phase-type distributions are called IFR and DFR-type is because it is possible to approximate a general IFR (DFR) distributed random variable B, by phase-type distributed random variables B_n of IFR-type (DFR-type). This will now be explained. Assume B is an IFR (DFR) distributed random variable. The approximating phase-type random variables, B_n, introduced in Subsection 4.1.3 can be fit into the framework of this section by choosing the following parameters. Choose $\mu_{i,k_i} = n$ for all k_i and $p^i_{0,1} = 1$, $p^i_{k_i,0} = \alpha_n(k_i)$, $p^i_{k_i,k_i+1} = 1 - \alpha_n(k_i)$ if $k_i < n^2$ and $p^i_{n^2,n^2} = 1 - \alpha_n(n^2)$. The failure rates are equal to $\mu_i(k_i) = \alpha_n(k_i)n$. Lemma 4.1.4 states that these failure rates are increasing (decreasing) in k_i. Hence B_n is indeed of IFR-type (DFR-type).

Instead of looking at the continuous-time process, we again focus on the uniformized chain. Assume $\nu = \lambda_0 + \lambda_1 + \lambda_2 + \mu_0 + \mu_1 + \mu_2 < \infty$, where $\mu_i = \sum_{k_i=1}^{f_i} \mu_{i,k_i}$. Let $\{\mathbf{N}_m\}$ be the embedded discrete-time Markov chain. Similar to Section 4.4, we will only show optimality results for the discrete-time process, since this automatically implies optimality for the continuous-time process.

In the remainder of this section we consider a linear network with two nodes (L=2) and derive properties a discipline should possess in order to achieve stochastic optimality. In Subsections 4.5.1 and 4.5.2 this is explicitly proved for phase-type distributions of IFR-type or DFR-type, for small and large class-0 users, respectively. In Subsection 4.5.3 the results are extended to general IFR and DFR distributions.

4.5.1 Small class-0 users

In this section we assume the service requirements are phase-type distributed. We first focus on the case that the class-0 users have uniformly greater failure rates than the sum of the failure rates of an arbitrary class-1 and 2 user, i.e. $\mu_0(k_0) \geq \mu_1(k_1) + \mu_2(k_2)$ for all k_0, k_1 and k_2. In the first lemma we show that if the service requirements of class 0 are either of IFR-type or DFR-type, then it is stochastically better to serve the "best" class-0 user whenever class 0 is present. By this we mean that every policy that does not always serve the "best" class-0 user can be stochastically improved by a policy that does. For IFR-type distributions, the "best" user is the user in the highest phase and for DFR-type distributions it is the user in the lowest phase.

Lemma 4.5.1 *If $\mu_0(k_0) \geq \mu_1(k_1) + \mu_2(k_2)$ for all k_0, k_1 and k_2, and the phase-type distributed service requirements of class 0 are of IFR-type or DFR-type, then it is stochastically better to serve the "best" class-0 user whenever Q_0 is non-empty.*

Remark 4.5.2 When within both classes 1 and 2 we can stochastically minimize the number of users present, we can completely characterize a stochastically optimal policy. For example if class i has a phase-type distribution of DFR-type, then within class i we should serve the user in the lowest phase. If class i has a phase-type distribution of IFR-type, then within class i we should serve the user in the highest phase. This can be proved in the same way as is done in [7] for the single-node case.

Proof of Lemma 4.5.1 Recall that we only have to prove the result for the embedded discrete-time Markov chain. The lemma will be proved by induction on the time horizon M. For $M = 1$ the result holds, since at most one departure is possible and the prescribed strategy maximizes the probability of a departure in case Q_0 is non-empty. Assume it holds for the time horizon $M - 1$. We will now consider a horizon of length M.
Suppose π is a policy that does not always serve the best class-0 user and that it cannot stochastically be improved. We will contradict this by constructing a policy π' that is stochastically better than π, i.e. $N_m^{\pi} \geq_{st} N_m^{\pi'}$ for all $m = 1, \ldots, M$, and therefore the induction result will hold for M as well. Note that after the first interval a horizon of length $M - 1$ remains. By induction we know that from then onward it is stochastically better to serve the "best" class-0 user whenever possible. Assume that in the first interval policy π does not obey the rule, hence Q_0 is non-empty. For the first interval we have to distinguish two cases: (i) serving classes 1 and 2 instead of class 0, and (ii) serving a class-0 user, but not the "best" one.

Case (i): First assume that in the first interval π serves classes 1 and/or 2. Tag these users as 1^* and 2^* (one of the users may not exist). At time $t = 0$ there is at least one class-0 user present. Choose one and tag it as 0^*.

To compare the behavior of two policies π and π', we use a coupling argument. Focusing on the uniformized processes, we define two types of events: arrival event (A); service event (S). A service event includes (dummy) phase transitions and (dummy) service completions. Dummy events are events of the uniformization process $\Lambda(t)$ that do not correspond to a transition in the original process, i.e. a self-transition. First we choose each interval between two events of the uniformized chain to be of type A or S with probabilities $\frac{\lambda_0+\lambda_1+\lambda_2}{\nu}$ and $\frac{\mu_0+\mu_1+\mu_2}{\nu}$, respectively. Also, in advance we generate the following four realization sequences: $\{U_i^A\}, \{U_i^*\}, \{U_i^0\}$ and $\{U_i^{12}\}$. The four realizations together are denoted as U. All realizations are independent and drawn from a uniform distribution on $[0, 1]$. The realizations are used in the following way. Given the type of the event and the user served in the interval we use one of the four realizations as shown in Table 4.2.

Type of event	User(s) served	Realization
A	any user(s)	$U_1^A, U_2^A, \ldots$
S	0^*	$U_1^*, U_2^*, \ldots$
S	class-0 user but not 0^*	$U_1^0, U_2^0, \ldots$
S	class-1 and/or 2 user	$U_1^{12}, U_2^{12}, \ldots$

Table 4.2: Realizations.

For example, when in an interval for the i-th time the type of the event is S and a class-0 user different from 0^* is served, the realization U_i^0 is used to determine whether or not there is a phase transition. If

$$\frac{\sum_{k_0=1}^{l_0-1}\mu_{0,k_0} + \sum_{k_0'=0}^{m_0-1} p^0_{l_0,k_0'}\mu_{0,l_0}}{\mu_0+\mu_1+\mu_2} \le U_i^0 < \frac{\sum_{k_0=1}^{l_0-1}\mu_{0,k_0} + \sum_{k_0'=0}^{m_0} p^0_{k_0,k_0'}\mu_{0,l_0}}{\mu_0+\mu_1+\mu_2},$$

then there is a (dummy) phase transition of a class-0 user from phase l_0 to phase m_0. If $m_0 = 0$, this means a (dummy) service completion of a class-0 user in phase l_0. If $\frac{\mu_0}{\mu_0+\mu_1+\mu_2} \le U_i^0 \le 1$, there is a dummy event. This way the process has the correct statistical properties, independent of which policy is used.

We now construct the policy π'. In the first interval it serves the 0^*-user. When the first event is of type A, it continues as π. When the first event is of type S, policy π' makes the same decisions as policy π would do under this realization, unless π would serve 0^*, in which case π' serves 1^* and 2^*. This is done until time σ. The random time σ is defined as the first time that a service event occurs while π' serves 1^* and 2^* or, equivalently, π serves 0^*. Note that until time σ, policy π will only serve class 0, because class 0 is always present up to time σ by definition of σ, and our induction hypothesis entails that π serves class 0 when present (after the first event). A realization of policies π and π' is shown in Figure 4.2.
Note that the value of U_1^{12} has no influence on the decisions of policy π at times $1 \le u < \sigma$, because by induction we know that policy π serves the "best" class-0 user, independent of which class-1 and

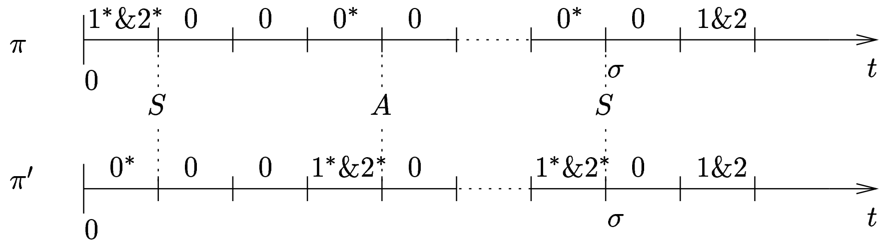

Figure 4.2: A realization of the two policies π and π'.

2 users are still present. For policy π', it is only important which class-0 user policy π serves. This can be observed by policy π' without anticipating. After time σ policy π' makes the same decisions as π would do in the same state. So policy π' is constructed without anticipating the future.

We want to stochastically compare the number of users in the system at a fixed time $m < M$ under the two policies π and π' for given realization sequences $\{U_i^A\}, \{U_i^*\}, \{U_i^0\}$ and $\{U_i^{12}\}$.

When the realization starts with an interval of type A, we compare policy π' with the behavior of policy π under the same four realization sequences and with the same realization of the types of the events in every interval. After the first interval, both policies are still in the same state. From that moment policy π' makes the same decisions as π. Hence, if the first interval is of type A, then $D_m^\pi = D_m^{\pi'}$, for all $m = 0, \ldots, M$, where D_m is the number of departed users at time m.

Let us now assume that the realization starts with an interval of type S. When U is such that $m < \sigma$, we compare policy π under the realization U, with policy π' under a corresponding realization $\overline{U}$. The types of event in each interval remain the same. The realization $\overline{U}$ is the same as U, except for $\overline{U}_1^*$. The sequence U is coupled with $\overline{U}$, such that if U_1^{12} prescribes a departure of either a class-1 or 2 user, then $\overline{U}_1^*$ prescribes a departure of user 0^*. This is possible because $\mu_0(k_0) \geq \mu_1(k_1) + \mu_2(k_2)$ for all k_0, k_1 and k_2. Note that σ is determined only by the sequence U_i^0 and the type of the events, which are common to both U and $\overline{U}$. We therefore have a bijection on the set of all realizations with $m < \sigma$.
By definition of $\overline{U}$, we have that if in the first interval a user (either 1^* or 2^*) leaves under π, then the 0^*-user leaves in the first interval under π'. Further we have that $D_{0,m}^\pi = D_{0,m}^{\pi'}$, where $D_{0,m}$ is the number of departed class-0 users, not counting the tagged user 0^* at time m. Before time m neither policy π nor π' serves classes 1 or 2 (except for 1^* and 2^*), so there is no possibility that they leave. In the interval $(1, m)$ there will be no departures of 0^*, 1^* or 2^* either, because $m < \sigma$. Therefore we can conclude that $D_m^\pi \leq D_m^{\pi'}$.

When U is such that $m \geq \sigma$, we compare policies π and π' by using the same four realization sequences and the same pattern of the type of event in every interval for both policies. At time σ both policies are in the same state and by definition of π', the two processes will proceed identically. Hence $D_m^\pi = D_m^{\pi'}$, for $m \geq \sigma$.

Case (ii): Now suppose that in the first interval policy π serves a class-0 user, but not the "best" one. Denote this user by 0^*. We will construct a policy π' that is stochastically better than π, which implies by induction that in the first interval it is optimal to serve the "best" class-0 user. Policy π' is constructed as follows. In the first interval π' serves the "best" class-0 user. When the first event is of type A, policy π' continues as π. When the first event is of type S, policy π' makes the same decisions as policy π would do, unless π serves the class-0 user that π' served already in the first interval, in which case π' serves 0^*. This is done until time σ, where σ is defined as the first time that a service event occurs while π' serves 0^*. Again we can use a coupling argument to compare the behavior of π and π' and derive that $D_m^\pi \leq_{st} D_m^{\pi'}$. This is similar to the proof in [7].

In both cases (i) and (ii), we have shown that there exists a policy π' such that $D_m^\pi \leq_{st} D_m^{\pi'}$, for all $m \leq M$. This is equivalent to $N_m^\pi \geq_{st} N_m^{\pi'}$, for all $m \leq M$, which was to be proved. □

Remark 4.5.3 In the proof of Lemma 4.5.1 we stochastically compared two policies π and π' by using a coupling argument. We showed that for every m there is a coupling between the two processes corresponding to π and π', such that at time m the stochastic ordering $N_m^{\pi'} \leq_{st} N_m^\pi$ holds. It was not possible to construct a coupling between these two processes, such that this order preserved for all m, i.e. $\{N_m^{\pi'}\} \leq_{st} \{N_m^\pi\}$. That such a coupling does not exist, is proved in [4].

4.5.2 Large class-0 users

In this section we assume the service requirements are phase-type distributed and focus on the case of large class-0 users. More precisely, the failure rates of class-0 users are uniformly smaller than the sum of the failure rates of an arbitrary class-1 and 2 user, i.e. $\mu_0(k_0) \leq \mu_1(k_1) + \mu_2(k_2)$ for all k_0, k_1 and k_2. In Lemma 4.5.4 we will partially characterize the stochastically optimal policy (assuming it exists). In order for that result to hold, we need to impose a restriction on the service requirement distributions of classes 1 and 2. They need to be of IFR-type. Later we will give a counterexample to show that this condition is necessary.

Lemma 4.5.4 *Assume $\mu_0(k_0) \leq \mu_1(k_1) + \mu_2(k_2)$ for all k_0, k_1 and k_2 and the service requirements of classes 1 and 2 are of IFR-type and $p_{0,1}^i = 1$, $i = 1, 2$, (so class-1 and 2 users always arrive in phase 1). Then it is stochastically better to serve the class-1 and 2 users in the highest phase, whenever there are users of both classes 1 and 2 present.*

Proof Recall that we only have to prove the result for the embedded discrete-time Markov chain. Again the lemma will be proved by induction on the time horizon M. For $M = 1$ the result holds, since at most one departure is possible and the prescribed strategy maximizes the probability of a departure in case classes 1 and 2 are both present. Assume it holds for the time horizon $M - 1$. We will now consider a horizon of length M.
Suppose π is a policy that does not always serve the class-1 and 2 users in the highest phase whenever there are users of both classes 1 and 2 present and that it cannot stochastically be improved. We will contradict this by constructing a policy π' that is stochastically better than π and therefore the induction result will be true for M as well. Note that after the first interval a horizon of length $M - 1$ remains. By induction we know that from then onward it is stochastically better to serve the class-1 and 2 users in the highest phase, whenever classes 1 and 2 are both present. Assume that in the first interval policy π does not obey the rule, hence all queues are non-empty. For the first interval we have two cases to distinguish: (i) serving class 0 instead of classes 1 and 2, and (ii) serving classes 1 and 2, but not the users in the highest phases.

Type of event	User(s) served	Realization
A	any user(s)	$U_1^A, U_2^A, \ldots$
S	0^*	$U_1^{0^*}, U_2^{0^*}, \ldots$
S	class-0 user but not 0^*	$U_1^0, U_2^0, \ldots$
S	1^* and 2^*	$U_1^{12^*}, U_2^{12^*}, \ldots$
S	class-1 and/or 2 user but not 1^* and 2^* together	$U_1^{12}, U_2^{12}, \ldots$

Table 4.3: Realizations.

Case (i): First assume that in the first interval π serves a class-0 user. Tag this user as 0^*. At time $t = 0$ there are class-1 and 2 users present. We denote the class-1 user in the highest phase by 1^* and the class-2 user in the highest phase by 2^*.

To compare the behavior of two policies π and π', we use a coupling argument. Again we define two types of events: arrival event (A); service event (S). First we choose each interval to be of type A or S with probabilities $\frac{\lambda_0+\lambda_1+\lambda_2}{\nu}$ and $\frac{\mu_0+\mu_1+\mu_2}{\nu}$, respectively. Also, in advance we generate the following five realization sequences: $\{U_i^A\}, \{U_i^{0^*}\}, \{U_i^0\}, \{U_i^{12^*}\}$ and $\{U_i^{12}\}$. The realizations are used as shown in Table 4.3. All realizations are independent and drawn from a uniform distribution on $[0, 1]$. The process has the correct statistical properties, independent of which policy is used.

We now construct the policy π'. In the first interval it serves the users 1^* and 2^*. When the first event is of type A, it continues as π. When the first event is of type S, policy π' starts serving 0^*. This is done until time σ. The random time σ is defined as the first time that a service event occurs after time 1. After time σ policy π' makes the same decisions as π would do in the same state. So policy π' is constructed without anticipating the future. In Figure 4.3 a realization of policies π and π' is shown.

We want to stochastically compare the number of users in the system at a fixed time $m < M$ under the two policies π and π'. We are given a realization, U, for which we will examine the behavior of policy π'.

When the realization starts with an interval of type A, we compare policy π' with the behavior of policy π under the same five realization sequences and with the same realization of the types of the events in every interval. After the first interval, both policies are still in the same state. From

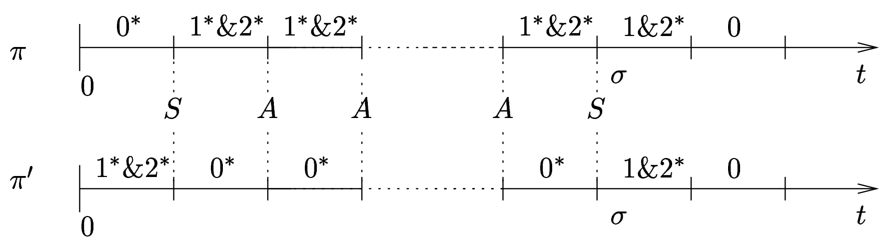

Figure 4.3: A realization of the two policies π and π'.

that moment policy π' makes the same decisions as π. Hence it is clear that $D_m^{\pi} = D_m^{\pi'}$, for all $m = 0, \ldots, M$.

Let us now assume that the realization starts with an interval of type S. When U is such that $m < \sigma$, we compare policy π under the realization U, with policy π' under a corresponding realization $\overline{U}$. The types of event in each interval remain the same. The realization $\overline{U}$ is the same as U, except for $\overline{U}_1^{12^*}$. The sequence $\overline{U}$ is coupled to U, such that if $U_1^{0^*}$ prescribes the departure of 0^*, then $\overline{U}_1^{12^*}$ prescribes a departure of either 1^* or 2^*. This is possible because $\mu_0(k_0) \leq \mu_1(k_1) + \mu_2(k_2)$ for all k_0, k_1 and k_2. Note that σ is determined only by the type of the events, which is common under both policies. We therefore have a bijection on the set of all realizations with $m < \sigma$.
By definition of $\overline{U}$, we have that if in the first interval 0^* leaves under π, then 1^* or 2^* leaves in the first interval under π'. In the remaining interval $(1, m)$ $(m < \sigma)$, by definition of σ there are no departures, hence $D_m^{\pi} \leq D_m^{\pi'}$.

When U is such that $m \geq \sigma$, we compare policies π and π' by using the same five realization sequences and the same pattern of the type of event in every interval for both policies. By definition of σ, by induction and because newly arriving class-1 and 2 users are in a lower phase, we know that policy π serves 1^* and 2^* in the interval $(1, \sigma)$. So at time σ both are in the same state. By definition of π', after time σ the two processes will proceed simultaneously. Hence $D_m^{\pi} = D_m^{\pi'}$, for $m \geq \sigma$.

Case (ii): Now suppose that in the first interval, π serves classes 1 and 2, but not the class-1 or 2 user in the highest phase. We can show in a similar way as before that there is a policy π' which is stochastically better.

In both cases we have shown that there exists a policy π' such that $D_m^{\pi} \leq_{st} D_m^{\pi'}$, for all $m \leq M$. This is equivalent to $N_m^{\pi} \geq_{st} N_m^{\pi'}$, for all $m \leq M$, which was to be proved. □

Remark 4.5.5 The restriction imposed on the distributions of classes 1 and 2 in Lemma 4.5.4 cannot be omitted, as is seen by the following example. In this example the service requirements of classes 0 and 1, B_0 and B_1, are exponentially distributed with means $1/\mu_0$ and $1/\mu_1$, respectively. The service requirement of class 2, B_2, is phase-type distributed with two phases. A class-2 user arrives in phase 1 which takes an exponentially distributed amount of time with mean $1/\mu_{2,1}$. With probability p it leaves the system and with probability $1-p$ it proceeds to phase 2. Phase 2 takes an exponentially distributed amount of time with mean $1/\mu_{2,2}$ after which the user leaves the system. We have the following numerical example. Choose the parameters $p = 0.8$ and $\lambda_0 = 1, \lambda_1 = 1, \lambda_2 = 3$ and $\mu_0 = 9, \mu_1 = 8, \mu_{2,1} = 10, \mu_{2,2} = 2$. The corresponding service requirements satisfy the conditions of Lemma 4.5.4, except that class 2 has a distribution of DFR-type. At time $t = 0$ the following users are present. There is one class-0 user, one class-1 user and one class-2 user which is in its second phase.
Define X^{π} as the number of users present in the system under policy π after five (dummy) events. We are interested in the probability that X^{π} is larger than 2, i.e. $\mathbb{P}(X^{\pi} > 2)$. Policy $\overline{\pi}$ is the minimizer of $\min_{\pi} \mathbb{P}(X^{\pi} > 2)$ and policy $\hat{\pi}$ is the policy that serves classes 1 and 2 whenever there are users of both classes present and otherwise it serves the class that will minimize $\mathbb{P}(X > 2)$. We calculated the probabilities under both policies. This resulted in $\mathbb{P}(X^{\hat{\pi}} > 2) = 0.4265$ and

$\mathbb{P}(X^{\overline{\pi}} > 2) = 0.4247$. We can conclude that in this case it is not necessarily stochastically better to serve classes 1 and 2 whenever there are users of both classes present, since policy $\overline{\pi}$ outperforms policy $\hat{\pi}$.
The policy $\overline{\pi}$ makes different decisions compared to $\hat{\pi}$. We observed that this only happens in situations in which there is a class-0 user, a class-1 user and a class-2 user in its second phase present. Policy $\hat{\pi}$ chooses to serve classes 1 and 2, while $\overline{\pi}$ chooses to serve class 0. Hence the optimal action in this state is to serve class 0, while users of classes 1 and 2 are present as well. This can be explained as follows. Serving classes 1 and 2 maximizes the output rate, but it has a negative effect as well. If only a class-2 user in phase 2 is present, then serving classes 1 and 2 carries the risk that class 1 becomes empty. When a new class-2 user arrives, it is not possible to work at the highest possible output rate. So there is a trade-off between maximizing the output rate at this moment or waiting for a new class-2 user and working later at the highest possible output rate.

Remark 4.5.6 If only users of classes 0 and 1 are present, we have not yet determined what the optimal action is. When $\mu_0(k_0) \geq \mu_1(k_1)$ for all k_0 and k_1, one would expect that it is always stochastically better to serve class 0. However, we did not succeed in proving this in the same way as we did in the previous two results, since we were not able to find a policy π' that is stochastically better than π, without anticipating the future.

4.5.3 General service requirements

In this subsection we return to generally distributed service requirements and in particular assume IFR or DFR distributed service requirements. Every positive random variable can be arbitrarily close approximated by a phase-type distributed random variable. In Section 4.1.3 a sequence of approximations was explicitly given. Denote by $B_{i,n}$ such a sequence that approximates the service requirement of class i, B_i. Suppose B_i is IFR (DFR) distributed, hence $B_{i,n}$ is of IFR-type (DFR-type). The service discipline has information available on the attained service of each user. The service requirement of a class-i user is approximated by $B_{i,n}$, where every phase takes an exponential time with mean $1/n$. For n large enough, knowing the attained amount of service of a class-i user is roughly equivalent with knowing in which phase that user is. We will assume that we know in which phase a user is. Recall from Section 4.1.3 that the failure rates of the random variable $B_{i,n}$, $\alpha_n(k_i)n$, behave as $r_i(\frac{k_i-1}{n})$, where r_i is the failure rate function of B_i. Hence if an ordering applies to the failure rate functions of $B_0, \ldots, B_L$, the ordering also applies to the failure rates of $B_{0,n}, \ldots, B_{L,n}$, for n large enough. The results derived in the previous section can therefore be generalized to IFR or DFR distributed service requirements.
Also note that for general distributions, the "best" user is the user with the highest failure rate. Therefore, if B_i is IFR it will always be optimal within class i to serve in a non-preemptive manner and if B_i is DFR it will always be optimal within class i to serve the users with the least attained service, which will result in the LAS discipline.

Chapter 5

Switching curve

In this chapter we return to exponential service requirements. In Section 4.4 the stochastically optimal policy was determined for certain combinations of the service rates. Only one situation was not treated yet, namely when there exists an $j = 1, \ldots, L$, such that $\sum_{i=1, i \neq j}^{L} \mu_i \geq \mu_0$. In this chapter this situation will be considered. A stochastically optimal policy may in general not exist, so instead we focus on the expected average optimal policy, i.e. the policy that minimizes $\mathbb{E}(N^\pi)$ over all policies π in Π. Recall that Π was defined as the set of policies that have no knowledge available of the remaining service requirements.

In this chapter we only consider the linear network with two nodes and hence focus on service rates such that $\mu_0 < \mu_i$ for at least one $i = 1, 2$. Intuitively it is clear that when there are users of both classes 1 and 2 present, serving them will be optimal. When there are only users of classes 0 and 1 present and $\mu_1 < \mu_0$, serving class 0 seems appropriate, since it is work conserving in both nodes and it maximizes the total output rate. However, when $\mu_0 < \mu_1$, we have no obvious rule which class to serve. Serving class 1 will maximize the total output rate of the system at this moment, but leaves node 2 unused. In contrast, serving class 0 is work conserving, but the total output rate of the system is not maximized. It seems reasonable that in situations in which there are only users of classes 0 and 1 present, there is a switching curve that determines which class is optimal to serve, that is for a given number of class-0 users, class 1 is served if the number of class-1 users is above a certain threshold and likewise for a given number of class-1 users, class 0 is served if the number of class-0 users is above a certain threshold. We will refer to such a policy as a policy with a switching curve structure.

In Section 5.1 we present some numerical results. This already supports the idea of a switching curve and gives some intuition on how the optimal policy is affected when changing the arrival or service rates. In Section 5.2 the expected average optimal policy is partially characterized. Using dynamic programming it is proved that in our model a switching curve exists. Finally in Section 5.3 we investigate a related fluid model which can be seen as the limit of our model considered at a large time scale. In this case an optimal policy is completely characterized for any choice of the service rates. In particular an explicit expression for the switching curve is derived.

5.1 Numerical results

We conducted numerical experiments on the linear network with two nodes, using the value iteration algorithm. In our model the state space, $\{(n_0, n_1, n_2) : n_0, n_1, n_2 = 0, 1, \ldots\}$, is infinite, since there is no restriction on the maximum number of users in the network. To determine the expected average optimal policy numerically, we need a finite state space. Therefore we truncate the state space by only allowing a maximum number of $B < \infty$ class-i users in the system (for $i = 0, 1, 2$). This means that we consider the same model, except class-i users that arrive when there are already B class-i users present, are rejected.

Truncating the state space influences the optimal actions. For example, consider the situation when there are already B class-0 users present. A newly arriving class-0 user is lost and we do not have to serve it. Since we do not assign costs to lost users, having B class-0 users in the system has a positive effect. It is now possible that the optimal action is to serve class 1 or class 2, while in the original system it might be better to serve class 0 in this state. Hence, around the boundaries of the state space, the optimal actions are influenced by these effects. In order to approximate the original model, we truncate at a sufficiently high level, i.e. B large enough, so that the states in the neighborhood of the truncation level will be visited with a very small probability.

We first present the optimal policies for three sets of parameters such that $\mu_1 \geq \mu_0$ and $\mu_2 \leq \mu_0$. These sets are shown in Table 5.1. We truncate at $B = 50$. We found that the optimal action when both Q_1 and Q_2 are non-empty, is to serve classes 1 and 2 and when only Q_0 and Q_2 are non-empty, the optimal action is to serve class 0. This is exactly what we expected, since for all three parameter sets we have $\mu_1 + \mu_2 \geq \mu_0$ and $\mu_2 \leq \mu_0$. Figures 5.1–5.3 show the optimal actions in the states when only Q_0 and Q_1 are non-empty for the three sets of parameters. A " $\circ$"-sign denotes that in that state it is optimal to serve class 0. We clearly see a switching curve. Below the switching curve the optimal action is to serve class 0 and above the switching curve it is to serve class 1. Around the truncation level the switching curve disappears. This is caused by the truncation, since when we truncate at a higher level, the switching curve keeps the same structure except for the region around the boundary.

Since the switching curves in the three figures differ, we can conclude that the optimal policy will not only depend on the relations between μ_0, μ_1 and μ_2, but also on the values of $\mu_0, \mu_1, \mu_2, \rho_0, \rho_1$ and ρ_2. Consider Figures 5.1 and 5.2. The corresponding parameter sets, set 1 and set 2, differ in the values of the parameters of class 2. Both λ_2 and μ_2 are multiplied with a factor 1/4. The traffic load remains the same, only the class-2 users arrive less frequently and their sizes are larger, that is class 2 has a larger time scale. As a result, the switching curve shifts downward, which

	λ_0	λ_1	λ_2	μ_0	μ_1	μ_2
Set 1	1	1.2	2	6	8	4
Set 2	1	1.2	0.5	6	8	1
Set 3	1	2.4	2	6	16	4

Table 5.1: Sets of parameters.

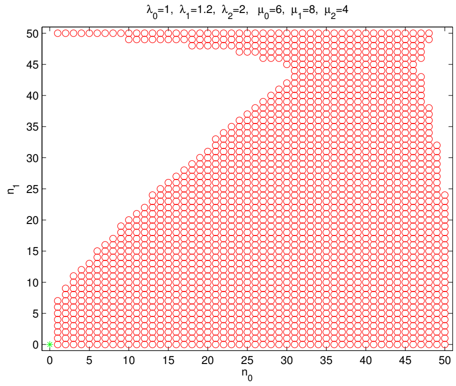

Figure 5.1: Optimal actions when Q_2 is empty, for set 1.

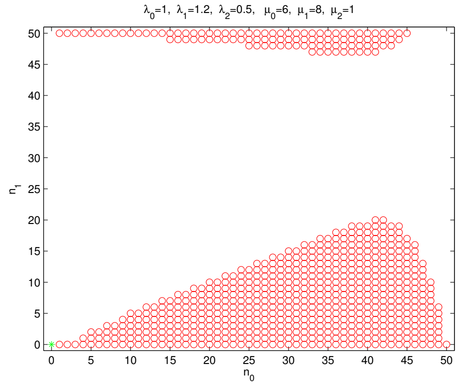

Figure 5.2: Optimal actions when Q_2 is empty, for set 2.

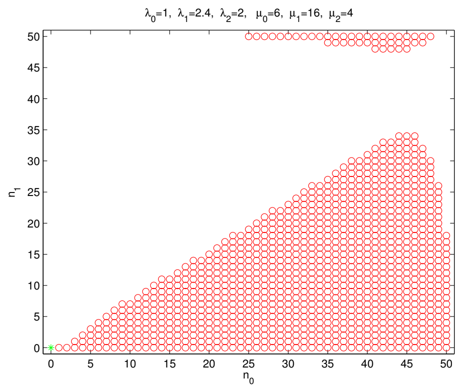

Figure 5.3: Optimal actions when Q_2 is empty, for set 3.

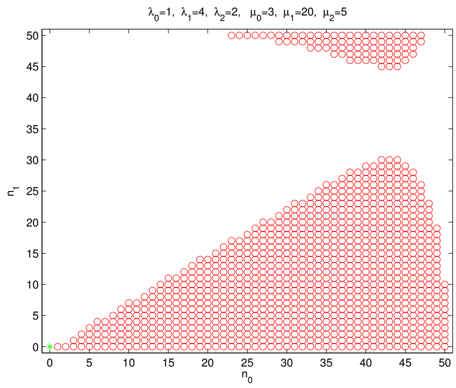

Figure 5.4: Optimal actions when Q_2 is empty.

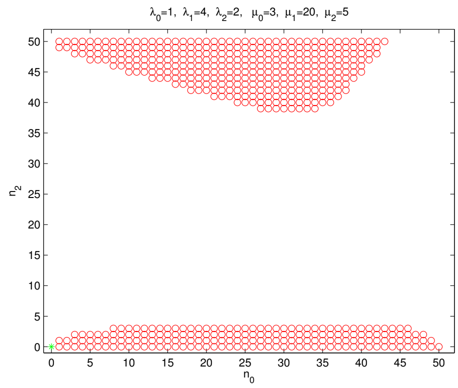

Figure 5.5: Optimal actions when Q_1 is empty.

implies that the policy decides already at a lower level to serve class 1 instead of class 0. This can be explained as follows. It now takes relatively longer before the first class-2 user arrives, so it is less promising to postpone serving class 1.
Set 1 and set 3 differ in the values of the parameters of class 1. Both λ_1 and μ_1 are multiplied with a factor 2. Again the traffic load remains the same, only the class-1 users arrive more frequently and their sizes are smaller, that is class 1 has a smaller time scale. As a result, the switching curve shifts downward, see Figures 5.1 and 5.3, which implies that the policy decides already at a lower level to serve class 1 instead of class 0. This can be explained as follows. When Q_2 is empty, serving class 1 becomes more attractive since this reduces the number of class-1 users relatively faster.

As a final numerical result we consider the situation that the parameters are such that $\mu_1, \mu_2 \geq \mu_0$. Now when Q_2 is empty, as well as when Q_1 is empty, there arises a switching curve. Figures 5.4 and 5.5 present the optimal policy for the set of parameters $\lambda_0 = 1, \lambda_1 = 4, \lambda_2 = 2, \mu_0 = 3, \mu_1 = 20$ and $\mu_2 = 5$.

5.2 Existence of a switching curve

In this section we will show that for the linear network with two nodes, the policy that minimizes the expected average number of users, $\mathbb{E}(N)$, can be characterized by a switching curve structure. To prove this we will use dynamic programming (DP). Again we only need to consider the embedded discrete-time Markov chain, which is denoted by $\{\mathbf{N}_m\}$. From now on we denote by i the number of class-0 users, by j the number of class-1 users and by k the number of class-2 users. The direct costs that are incurred each time state (i, j, k) is visited, are $c(i, j, k)$. Define by $V_m^\pi(n_0, n_1, n_2)$ the total expected costs in $0, \ldots, m-1$ when starting at time 0 in state $\mathbf{N}_0^\pi = (n_0, n_1, n_2)$ and using policy π, that is $V_m^\pi(n_0, n_1, n_2) = \mathbb{E}(\sum_{k=0}^{m-1} c(\mathbf{N}_k^\pi))$, with $\mathbf{N}_0^\pi = (n_0, n_1, n_2)$. Define $g^\pi = \lim_{m\to\infty} \mathbb{E}(c(\mathbf{N}_m^\pi))$. The value function V^π is defined as $V^\pi(n_0, n_1, n_2) = \lim_{m\to\infty}[V_m^\pi(n_0, n_1, n_2) - g^\pi \cdot m]$. We are interested in the policy that minimizes g.

Assume, without loss of generality, $\lambda_0 + \lambda_1 + \lambda_2 + \mu_0 + \mu_1 + \mu_2 = 1$. Using uniformization, the DP equation can be written as:

$$\begin{aligned} V_0(i,j,k) &= 0 \\ V_{n+1}(i,j,k) &= c(i,j,k) + \lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1) \\ &\quad + \min\{\mu_0 V_n((i-1)^+,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k), \\ &\quad \mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,(k-1)^+)\}, \quad \text{for} \quad n = 0,1,\ldots. \end{aligned}$$

A policy π minimizes g if the corresponding value function V^π satisfies the following property: the inequality $\mu_0 V^\pi((i-1)^+,j,k) + \mu_1 V^\pi(i,j,k) + \mu_2 V^\pi(i,j,k) \le \mu_0 V^\pi(i,j,k) + \mu_1 V^\pi(i,(j-1)^+,k) + \mu_2 V^\pi(i,j,(k-1)^+)$ holds if and only if π serves class 0 in state (i,j,k). This result can be found in for example [11].

Choosing $c(i,j,k) = i + j + k$ implies that the objective is to find a policy π that minimizes $g^\pi = \lim_{m\to\infty} \mathbb{E}(N_m^\pi)$. Hence, the policy π for which the value function V^π satisfies the property above, minimizes the expected average total number of users in the system. For now, we do not consider this particular choice of a cost function and derive, under certain conditions on $c(i,j,k)$, inequalities for the value function which partially characterize the optimal policy. To prove such inequalities, we will derive that the inequalities hold for V_n, for all n.

First we present two familiar results. Lemma 5.2.1 states that under certain conditions on $c(i,j,k)$, the optimal action is to always serve classes 1 and 2 whenever there are users of both classes present. Lemma 5.2.2 states that under certain conditions on $c(i,j,k)$ it is always better to serve class 0 rather than class i, $i = 1,2$ alone.

Lemma 5.2.1 *If $W = c$ is non-decreasing in i,j and k and satisfies for all $i,j,k > 0$,*

$$\mu_0 W(i,j,k) + \mu_1 W(i,j-1,k) + \mu_2 W(i,j,k-1) \le \mu_0 W(i-1,j,k) + (\mu_1 + \mu_2) W(i,j,k), \tag{5.1}$$

then (5.1) with $W = V_n$ holds for all n.

Lemma 5.2.2 *If $W = c$ is non-decreasing in i,j and k and satisfies for all $i,j > 0, k \ge 0$,*

$$\mu_0 W(i-1,j,k) + \mu_1 W(i,j,k) \le \mu_0 W(i,j,k) + \mu_1 W(i,j-1,k), \tag{5.2}$$

then (5.2) with $W = V_n$ holds for all n.
If $W = c$ is non-decreasing in i,j and k and satisfies for all $i,k > 0, j \ge 0$,

$$\mu_0 W(i-1,j,k) + \mu_2 W(i,j,k) \le \mu_0 W(i,j,k) + \mu_2 W(i,j,k-1), \tag{5.3}$$

then (5.3) with $W = V_n$ holds for all n.

The proof of Lemmas 5.2.1 and 5.2.2 proceeds along similar lines as in Lemmas 4.4.4 and 4.4.3 and will be omitted.

Now we derive inequalities which imply that an optimal policy has a switching curve structure. The existence of a switching curve when there are no class-2 users present is equivalent to the value function, V, satisfying Properties 1 and 2 below. By symmetry, similar properties hold for the existence of a switching curve when there are no class-1 users.

Property 1: If it is optimal to serve class 1 in state $(i, j, 0)$, then this is optimal in state $(i, j+1, 0)$ as well, or equivalently, if

$$\mu_0 V(i,j,0) + \mu_1 V(i,j-1,0) + \mu_2 V(i,j,0) \leq \mu_0 V(i-1,j,0) + \mu_1 V(i,j,0) + \mu_2 V(i,j,0),$$

then

$$\mu_0 V(i,j+1,0) + \mu_1 V(i,j,0) + \mu_2 V(i,j+1,0) \leq \mu_0 V(i-1,j+1,0) + \mu_1 V(i,j+1,0) + \mu_2 V(i,j+1,0).$$

Note that this property is implied by the following inequality:

$$\begin{aligned} &\mu_0 V(i,j+1,0) + \mu_0 V(i-1,j,0) + 2\mu_1 V(i,j,0) \\ &\leq \mu_0 V(i,j,0) + \mu_0 V(i-1,j+1,0) + \mu_1 V(i,j-1,0) + \mu_1 V(i,j+1,0). \end{aligned} \tag{5.4}$$

Property 2: If it is optimal to serve class 0 in state $(i, j, 0)$, then this is optimal in state $(i+1, j, 0)$ as well, or equivalently, if

$$\mu_0 V(i-1,j,0) + \mu_1 V(i,j,0) + \mu_2 V(i,j,0) \leq \mu_0 V(i,j,0) + \mu_1 V(i,j-1,0) + \mu_2 V(i,j,0),$$

then

$$\mu_0 V(i,j,0) + \mu_1 V(i+1,j,0) + \mu_2 V(i+1,j,0) \leq \mu_0 V(i+1,j,0) + \mu_1 V(i+1,j-1,0) + \mu_2 V(i+1,j,0).$$

This property is implied by the following inequality:

$$\begin{aligned} &2\mu_0 V(i,j,0) + \mu_1 V(i+1,j,0) + \mu_1 V(i,j-1,0) \\ \leq\ &\mu_0 V(i+1,j,0) + \mu_0 V(i-1,j,0) + \mu_1 V(i+1,j-1,0) + \mu_1 V(i,j,0). \end{aligned} \tag{5.5}$$

To derive the main result of a switching curve structure, we will show that inequalities (5.4) and (5.5) and the two analogous versions of them (when class 1 is empty) hold for V. The next lemma states in particular that these four inequalities, as well as three other inequalities, are satisfied for V_n, $n = 0, 1, \ldots$. The three inequalities are helpful in proving the former four inequalities. The proof of Lemma 5.2.3 may be found in Appendix A.

Lemma 5.2.3 *If $W = c$ satisfies (5.1) as well as the following four inequalities, for all $i > 0, j \geq 0, k \geq 0$,*

$$\begin{aligned} &\mu_0 W(i,j+1,k) + \mu_0 W(i-1,j,k) + 2\mu_1 W(i,j,k) \\ &\leq \mu_0 W(i,j,k) + \mu_0 W(i-1,j+1,k) + \mu_1 W(i,(j-1)^+,k) + \mu_1 W(i,j+1,k), \end{aligned} \tag{5.6}$$

$$\begin{aligned} &\mu_0 W(i,j,k+1) + \mu_0 W(i-1,j,k) + 2\mu_2 W(i,j,k) \\ &\leq \mu_0 W(i,j,k) + \mu_0 W(i-1,j,k+1) + \mu_2 W(i,j,(k-1)^+) + \mu_2 W(i,j,k+1), \end{aligned} \tag{5.7}$$

$$\begin{aligned} &2\mu_0 W(i,j,k) + \mu_1 W(i+1,j,k) + \mu_1 W(i,(j-1)^+,k) \\ &\leq \mu_0 W(i+1,j,k) + \mu_0 W(i-1,j,k) + \mu_1 W(i+1,(j-1)^+,k) + \mu_1 W(i,j,k), \end{aligned} \tag{5.8}$$

$$\begin{aligned} &2\mu_0 W(i,j,k) + \mu_2 W(i+1,j,k) + \mu_2 W(i,j,(k-1)^+) \\ &\leq \mu_0 W(i+1,j,k) + \mu_0 W(i-1,j,k) + \mu_2 W(i+1,j,(k-1)^+) + \mu_2 W(i,j,k), \end{aligned} \tag{5.9}$$

and the following three inequalities, for all $i \geq 0, j \geq 0, k \geq 0$,

$$\begin{aligned} W(i,j,(k-1)^+) + W((i-1)^+,j,k) &\leq W(i,j,k) + W((i-1)^+,j,(k-1)^+), && (5.10)\\ W(i,(j-1)^+,k) + W((i-1)^+,j,k) &\leq W(i,j,k) + W((i-1)^+,(j-1)^+,k), && (5.11)\\ W(i,(j-1)^+,(k-1)^+) + W(i,j,k) &\leq W(i,(j-1)^+,k) + W(i,j,(k-1)^+), && (5.12)\end{aligned}$$

then (5.6)–(5.12) with $W = V_n$ hold for all n.

By definition of the value function we can write

$$\lim_{n\to\infty}\Big(V_n(i,j,k) - V_n(0,0,0)\Big) = \lim_{n\to\infty}\Big(V_n(i,j,k) - g\cdot n - V_n(0,0,0) + g\cdot n\Big) = V(i,j,k) - V(0,0,0).$$

So if for all V_n one of the inequalities is satisfied, then that is the case for V as well. Further note that $c(i,j,k) = i+j+k$ satisfies all conditions of Lemmas 5.2.1–5.2.3 if $\mu_1+\mu_2 \geq \mu_0$, which implies that the corresponding inequalities are satisfied for all V_n and hence for V. In particular, when choosing $k = 0$ in (5.6) and (5.8) and $j = 0$ in (5.7) and (5.9), we obtain the desired inequalities for the existence of a switching curve. This results in the following corollary.

Corollary 5.2.4 *If $\mu_1+\mu_2 \geq \mu_0$, then the expected average optimal stationary policy serves classes 1 and 2 whenever both Q_1 and Q_2 are non-empty. Otherwise the optimal policy has a switching curve structure.*

5.3 Fluid model

In the previous section we established the existence of a switching curve in the case of two nodes and exponential service requirements. A parametric characterization of the curve could not be given. In this section we will consider a fluid model for which we can derive optimal policies and in particular obtain the exact closed-form expression for the switching curve. This is motivated by work of several researchers who studied the optimal scheduling of fluid limits of stochastic networks, see for example [1, 13].

Throughout this section we only consider service rates such that $\mu_1 + \mu_2 \geq \mu_0$, since then a switching curve may occur in the original model. The fluid model can be seen as a limiting case of the original model, considered at a large time scale. The relation between the original model and the fluid model will be explained in Subsection 5.3.1. In Subsection 5.3.2 we present and prove the optimal policies for various choices of the service rates and derive an analytical expression for the switching curve. In Subsection 5.3.3 we evaluate the optimal policies for the fluid model and discuss their performance in the original model.

5.3.1 Fluid model based on the original model

In the original system class-i users arrive with rate λ_i and are served with rate μ_i. Now consider a sequence of systems, indexed by ϵ, where in the ϵ-system the arrival and service rates of class i are $\frac{\lambda_i}{\epsilon}$ and $\frac{\mu_i}{\epsilon}$, respectively. Hence the traffic loads of the classes remain the same. Denote by $n_i(t)$ and $\tilde{y}_i^\epsilon(t)$ the number of class-i users in the original system and in the ϵ-system, respectively. The dynamics of the ϵ-system at time t are related to the dynamics of the original system at the larger time scale t/ϵ, in particular $\tilde{y}_i^\epsilon(t) \approx n_i(t/\epsilon)$ for ϵ small enough. So when we consider the original system at a large time scale, we can equivalently study the ϵ-system for ϵ small enough. We further define the scaled number of class-i users in the ϵ-system by $y_i^\epsilon(t) := \tilde{y}_i^\epsilon(t)\epsilon$.

In the limit as $\epsilon \downarrow 0$, the ϵ-system behaves as a fluid model where work flows into Q_i at rate ρ_i and drains at rate $s_i(t)$ at time t. The value $s_i(t)$ is the fraction of capacity that is allocated to class i at time t and satisfies $s_0(t) + s_i(t) \leq 1$, for $i = 1, 2$. Denote by $w_i(t)$ the amount of class-i work in the fluid model at time t and $\boldsymbol{w}(t) = (w_0(t), w_1(t), w_2(t))$. The behavior of the work in the fluid model can be described as follows:

$$\begin{aligned} \dot{w}_i(t) &= \rho_i - s_i(t), \text{ for } i = 0, 1, 2, \\ w_i(t) &\geq 0, \text{ for } i = 0, 1, 2, \\ s_0(t) + s_i(t) &\leq 1, \text{ for } i = 1, 2, \\ s_i(t) &\geq 0, \text{ for } i = 0, 1, 2, \\ \boldsymbol{w}(0) &= \boldsymbol{w}, \end{aligned}$$

where we used the notation $\dot{x}(t) = \frac{\mathrm{d}x}{\mathrm{d}t}(t)$. Note that the evolution of the fluid model involves no stochasticity, which renders it more tractable. We denote by $y_i(t) := \lim_{\epsilon \downarrow 0} y_i^\epsilon(t)$ the "length" of the fluid queue in the fluid model at time t. This quantity behaves as $y_i(t) = w_i(t)\mu_i$, since $\tilde{y}_i^\epsilon(t) \sim w_i(t)\frac{\mu_i}{\epsilon}$ as $\epsilon \downarrow 0$ and $y_i(t) = \lim_{\epsilon \downarrow 0} \tilde{y}_i^\epsilon(t)\epsilon$.

This section will study the above-described fluid model. In particular we derive the most optimal policy to return to the origin, starting from a given initial state. This corresponds to the original system when we start with a very large number of users and we want to empty the system as optimally as possible.

5.3.2 Optimal policies

In the fluid model, a policy decides how much capacity is allocated to each class, that is a policy determines $s_0(t), s_1(t)$ and $s_2(t)$. Let Π be the set of all policies that satisfy $s_0(t) + s_i(t) \leq 1$, for $i = 1, 2$, $s_i(t) \leq \rho_i$ when $w_i(t) = 0$ and $s_i(t) \geq 0$ for $i = 0, 1, 2$. We seek a policy that is optimal. We use the following two definitions for an optimal policy.

- Policy $\overline{\pi}$ is called path-wise optimal if $\sum_{i=0}^2 y_i^{\overline{\pi}}(t) \leq \sum_{i=0}^2 y_i^\pi(t)$ for all $t \geq 0$ and for all $\pi \in \Pi$.

Path-wise optimal policies do not necessarily exist. Therefore we have a second criterion.

- Policy $\overline{\pi}$ is called average optimal if $\int_0^T \sum_{i=0}^2 y_i^{\overline{\pi}}(t)\mathrm{d}t \leq \int_0^T \sum_{i=0}^2 y_i^\pi(t)\mathrm{d}t$ for all $\pi \in \Pi$, with T such that the system is empty at time T.

We see that a path-wise optimal policy, if it exists, is automatically average optimal. Note that the definitions can also be expressed in terms of workload, using $y_i(t) = \mu_i w_i(t)$.

In node i, exactly $1 - s_i(t)$ is left for class 0, for $i = 1, 2$, but since class 0 needs both servers at the same time, $s_0(t) \leq 1 - s_i(t)$ for $i = 1, 2$. It is clear that it cannot be optimal to allocate the capacity of the nodes such that more capacity could have been given to a user, without decreasing the obtained fraction of capacity for the other users. So $s_0(t) = 1 - s_i(t)$ for at least one i, say $i = 2$. It is however still possible that $s_0(t) < 1 - s_1(t)$, i.e. capacity is lost in node 1 at time t. This occurs when $w_1(t) = 0$ and $s_2(t) > \rho_1$. For example, if $w_2(t) > 0$ and the policy gives the total capacity to classes 1 and 2, then $s_1(t) = \rho_1$ and $s_2(t) = 1$. Hence $s_0(t) = 0$ and the wasted capacity at node 1 is equal to $1 - \rho_1$. Therefore the fluid model retains the non-work conserving property of the original model.

From now on we will assume without loss of generality $\rho_1 \le \rho_2$ and $\rho_0 + \rho_i < 1$ for $i = 1, 2$. The latter condition is needed for stability and guarantees that the fluid model will empty at some point and from then on remains empty if controlled optimally. In our fluid model, either a path-wise or an average optimal policy exists. Proposition 5.3.1 presents these policies.

Proposition 5.3.1 *Assume $\rho_1 \le \rho_2$, $\rho_0 + \rho_i < 1$ for $i = 1, 2$ and $\mu_1 + \mu_2 \ge \mu_0$. A sample-path optimal policy can be found in the following situations:*

Case	*Sample-path optimal policy*	
$\mu_0 \ge \mu_1, \mu_2$	$s_0 = 0$	*if* $w_1, w_2 > 0$
	$s_0 = 1 - \rho_2 - \mathbf{1}_{(w_0=0)}(1 - \rho_0 - \rho_2)$	*if* $w_1 > 0, w_2 = 0$
	$s_0 = 1 - \rho_1 - \mathbf{1}_{(w_0=0)}(1 - \rho_0 - \rho_1)$	*if* $w_1 = 0$
$\mu_2 \ge \mu_0 \ge \mu_1$	$s_0 = 0$	*if* $w_2 > 0$
	$s_0 = 1 - \rho_2 - \mathbf{1}_{(w_0=0)}(1 - \rho_0 - \rho_2)$	*if* $w_2 = 0$

An average optimal policy can be found in the following situations:

Case	*Average optimal policy*	
$\mu_1 \ge \mu_0 \ge \mu_2$	$s_0 = 1 - \rho_1 - \mathbf{1}_{(w_0=0)}(1 - \rho_0 - \rho_1)$	*if* $w_1 = 0$
	$s_0 = 1 - \rho_2$	*if* $w_1 > 0$, $w_2 = 0$ *and* $w_1 \le \frac{\mu_2}{\mu_1+\mu_2-\mu_0}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}w_0$
	$s_0 = 0$	*if* $w_1 > 0$, $w_2 = 0$ *and* $w_1 \ge \frac{\mu_2}{\mu_1+\mu_2-\mu_0}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}w_0$
	$s_0 = 0$	*if* $w_1, w_2 > 0$
$\mu_1, \mu_2 \ge \mu_0$	$s_0 = 1 - \rho_2 - \mathbf{1}_{(w_0=0)}(1 - \rho_0 - \rho_2)$	*if* $w_2 = 0$ *and* $w_1 \le \frac{\mu_0}{\mu_1}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}w_0$
	$s_0 = 0$	*if* $w_1 > 0$, $w_2 = 0$ *and* $w_1 \ge \frac{\mu_0}{\mu_1}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}w_0$
	$s_0 = 0$	*if* $w_2 > 0$.

In all cases $s_i = 1 - s_0$ if $w_i > 0$ and $s_i = \min(\rho_i, 1 - s_0)$ if $w_i = 0$, for $i = 1, 2$.

Remark 5.3.2 The optimal policies for the two cases $\mu_1 \ge \mu_0 \ge \mu_2$ and $\mu_1, \mu_2 \ge \mu_0$, both involve a switching curve when $w_2 = 0$. Note that the switching curve is in fact linear. When (w_0, w_1) lies above the curve, i.e. the ratio w_1/w_0 exceeds a certain threshold, we choose not to serve class 0. However, when (w_0, w_1) lies below the curve, class 0 receives the fraction $1 - \rho_2$ that is left from keeping class 2 empty.
When $\mu_1 \ge \mu_0 \ge \mu_2$, the switching curves expressed in the length of the queues is given by

$$y_1 = \frac{\mu_2}{\mu_1 + \mu_2 - \mu_0}\frac{\mu_1}{\mu_0}\frac{\rho_2 - \rho_1}{1 - \rho_0 - \rho_2}y_0.$$

This shows that the curve depends on the traffic loads as well as on the service rates.
When $\mu_1, \mu_2 \ge \mu_0$ the switching curve is given by

$$y_1 = \frac{\rho_2 - \rho_1}{1 - \rho_0 - \rho_2}y_0,$$

hence it now only depends on the traffic loads.

The remainder of this subsection will be used to prove Proposition 5.3.1. It will be useful to introduce the following notation. Suppose $\boldsymbol{w}^1, \boldsymbol{w}^2 \in \mathbb{R}_+^3$ and $\boldsymbol{w}(0) = \boldsymbol{w}^1$. Denote by $T^\pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ the time it takes under policy π to move from $\boldsymbol{w}^1$ to $\boldsymbol{w}^2$, i.e. $\boldsymbol{w}^\pi(T^\pi(\boldsymbol{w}^1, \boldsymbol{w}^2)) = \boldsymbol{w}^2$. Let $K^\pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ denote the costs of moving from $\boldsymbol{w}^1$ to $\boldsymbol{w}^2$ under policy π, so $K^\pi(\boldsymbol{w}^1, \boldsymbol{w}^2) := \int_0^{T^\pi(\boldsymbol{w}^1,\boldsymbol{w}^2)} \sum_{i=0}^2 \mu_i w_i^\pi(t)\mathrm{d}t$. Denote by $K(\boldsymbol{w}^1, \boldsymbol{w}^2)$ the minimum cost of moving from $\boldsymbol{w}^1$ to $\boldsymbol{w}^2$, i.e.

$$K(\boldsymbol{w}^1, \boldsymbol{w}^2) := \min_{\pi \in \Pi} K^\pi(\boldsymbol{w}^1, \boldsymbol{w}^2).$$

The first observation we make concerns the situation when there is work of both classes 1 and 2 present. For the original model it was proved in the previous chapter that it is optimal to serve both classes 1 and 2. The fluid model inherits this property. We state the following observation without proof.

Observation 5.3.3 (Classes 1 and 2 both backlogged) *Assume $\mu_1 + \mu_2 \geq \mu_0$. Suppose at time t the state is $\boldsymbol{w}(t) = \boldsymbol{w}$. When work of both classes 1 and 2 is present, an optimal policy allocates the capacity of each node completely to classes 1 and 2. Hence if $w_1, w_2 > 0$, then $s_1(t) = s_2(t) = 1$.*
Furthermore, when there is no backlog of either class 1 or class 2, an optimal policy always keeps at least one of these classes empty. Hence, if $w_i = 0$ and $w_j > 0$, $i, j = 1, 2$, then $s_i(t) = \rho_i$ and $s_j(t) \geq \rho_i$. If $w_i = w_j = 0$ and $\rho_i \leq \rho_j$, then $s_i(t) = \rho_i$ and $s_j(t) \geq \rho_i$.

Observation 5.3.3 fully characterizes the optimal policy in states where both classes 1 and 2 are backlogged. In the remainder of this section we therefore only need to consider the following two cases: (1) no backlog of class 1 and (2) no backlog of class 2.

Case 1: no backlog of class 1

Suppose at time t the system is in state $\boldsymbol{w}(t) = \boldsymbol{w}$, with $w_1 = 0$. Observation 5.3.3 implies that in the $w_1 = 0$-plane we have $s_1(t) = \rho_1$. Hence once the optimal trajectory has entered the $w_1 = 0$-plane, it will stay in this plane from then on. Class 2 receives at least capacity ρ_1, i.e. $s_2(t) \geq \rho_1$. We seek a path-wise optimal policy, that is we need to determine the optimal value for $s_2(t)$.
Observe that as long as $w_0 + w_2 > 0$ the optimal policy is work conserving in node 2. Even when $w_2 = 0$, since then $s_2(t)$ is such that $s_1(t) = \rho_1 \leq s_2(t) \leq \rho_2$ and $s_0(t) = \min(1 - s_1(t), 1 - s_2(t)) = 1 - s_2(t)$. Since we do not leave the $w_1 = 0$-plane and every policy is work conserving in node 2, the time until reaching the origin is known in advance. It is equal to $T(\boldsymbol{w}, \boldsymbol{0}) = \frac{w_0 + w_2}{1 - \rho_0 - \rho_2}$, because the total work in node 2 is equal to $w_0 + w_2$ and the continuous rate at which it decreases is $1 - \rho_0 - \rho_2$. Let $y(t)$ denote the total length of the fluid queue at time t, i.e. $y(t) = y_0(t) + y_1(t) + y_2(t)$. The change in the number of users at time t can be written as:

$$\mathrm{d}y(t) = (\lambda_0 + \lambda_2 - (1 - s_2(t))\mu_0 - s_2(t)\mu_2) \cdot \mathrm{d}t, \quad \text{for } \mathrm{d}t \text{ small},$$

with $s_2(t)$ such that $y_i(t) \geq 0$, for $i = 0, 2$ and $s_2(t) \geq \rho_1$. In node 2, at least ρ_1 is given to class 2, which leaves a capacity of $1 - \rho_1$ still to be allocated among classes 0 and 2. Since every policy reaches the origin at the same time, it is path-wise optimal to operate at the highest possible service rate. The class with the highest service rate is therefore given the left-over capacity $1 - \rho_1$. We distinguish the two scenarios and give the corresponding path-wise optimal policies.

- $\mu_0 \le \mu_2$: Choose $s_1(t) = \rho_1, s_0(t) = (1-\rho_2)\mathbf{1}_{(w_2(t)=0)}$ and $s_2(t) = 1-(1-\rho_2)\mathbf{1}_{(w_2(t)=0)}$, so class 2 receives priority.
- $\mu_0 \ge \mu_2$: Choose $s_1(t) = \rho_1$, $s_0(t) = 1-\rho_1-(1-\rho_0-\rho_1)\mathbf{1}_{(w_0(t)=0)}$ and $s_2(t) = \rho_1 + (1-\rho_0-\rho_1)\mathbf{1}_{(w_0(t)=0)}$, so class 0 receives priority.

The optimal paths for several starting points are drawn in Figure 5.6 for $\mu_0 \le \mu_2$ and in Figure 5.7 for $\mu_0 \ge \mu_2$.

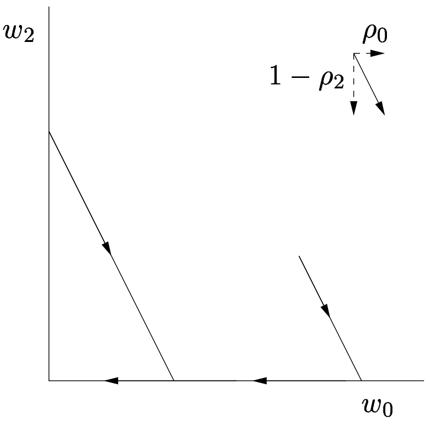

Figure 5.6: Path-wise optimal trajectories in the $w_1 = 0$-plane if $\mu_0 \le \mu_2$.

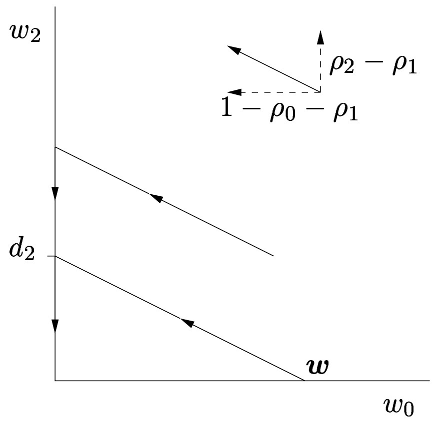

Figure 5.7: Path-wise optimal trajectories in the $w_1 = 0$-plane if $\mu_0 \ge \mu_2$.

We will now determine the costs, $K(\boldsymbol{w}, \mathbf{0})$, in case $\boldsymbol{w} = (w_0, 0, 0)$, since this will be useful in the next derivation.

- First let $\mu_0 \le \mu_2$. It takes $T(\boldsymbol{w}, \mathbf{0}) = \frac{w_0}{1-\rho_0-\rho_2}$ to reach the origin. Under the optimal policy there is no backlog of class 2. The length of the fluid queue of class 0 decreases linear and the average length is $w_0/2$. Hence

$$K(\boldsymbol{w}, \mathbf{0}) = T(\boldsymbol{w}, \mathbf{0})\mu_0 \frac{w_0}{2} = \frac{1}{2}\frac{\mu_0 w_0^2}{1-\rho_0-\rho_2}, \tag{5.13}$$

with $\boldsymbol{w} = (w_0, 0, 0)$.

- Now let $\mu_0 \ge \mu_2$. An optimal trajectory will look like the path in Figure 5.7. It starts in $\mathbf{w}$ and it hits the vertical axis in $\boldsymbol{d} = (0, 0, d_2)$. The time it takes to empty the system of class 0 is equal to $T(\boldsymbol{w}, \boldsymbol{d}) = \frac{w_0}{1-\rho_0-\rho_1}$. At that time the total work of class 2 has increased from 0 to d_2 and after that it decreases again to 0. Note that d_2 is equal to $d_2 = T(\boldsymbol{w}, \boldsymbol{d})(\rho_2 - \rho_1) = w_0 \frac{\rho_2-\rho_0}{1-\rho_0-\rho_1}$. We can conclude that the costs are equal to

$$\begin{aligned} K(\boldsymbol{w}, \mathbf{0}) &= T(\boldsymbol{w}, \boldsymbol{d})\mu_0 \frac{w_0}{2} + T(\boldsymbol{w}, \mathbf{0})\mu_2 \frac{d_2}{2} \\ &= \frac{1}{2} w_0^2 \Big(\frac{\mu_0}{1-\rho_0-\rho_1} + \frac{\mu_2(\rho_2-\rho_1)}{(1-\rho_0-\rho_1)(1-\rho_0-\rho_2)} \Big), \end{aligned} \tag{5.14}$$

with $\boldsymbol{w} = (w_0, 0, 0)$.

Case 2: no backlog of class 2

Now assume that at time t the system is in state $\boldsymbol{w}(t) = \boldsymbol{w}$, with $w_2 = 0$. It can be checked that the policy that minimizes the time it takes to reach the origin, $T^\pi(\boldsymbol{w}, \boldsymbol{0})$, is the policy that gives class 0 as much capacity as possible until $w_1 = 0$ and from then on, every work-conserving policy takes the same amount of time. We now distinguish the following two cases:

- $\mu_0 \geq \mu_1$: The above-described shortest trajectory to the origin is exactly the path that, given the restriction of Observation 5.3.3, maximizes the output rate until the process hits the horizontal axis, i.e. $w_1 = 0$. Therefore this trajectory minimizes the term $\sum_{i=0}^{2} y_i(t) = \sum_{i=0}^{2} \mu_i w_i(t)$, for every t, until it hits the horizontal axis. From then on we are in Case 1, for which we already know there exists a sample-path optimal policy. Hence there exists a sample-path optimal policy. When $w_1(t) > 0$ and $w_2(t) = 0$ it is as follows: $s_0(t) = 1 - \rho_2 - \mathbf{1}_{(w_0(t)=0)}(1 - \rho_0 - \rho_2)$, $s_1(t) = \rho_2 + \mathbf{1}_{(w_0(t)=0)}(1 - \rho_0 - \rho_2)$ and $s_2(t) = \rho_2$.

- $\mu_0 \leq \mu_1$: In this case the above-described shortest trajectory is certainly not path-wise optimal. In contrast to the path-wise optimal policies derived earlier, only an average optimal policy can be found. In the remainder of this section we will further investigate this case.

Suppose $\boldsymbol{w} = (w_0, w_1, 0)$, with $w_1 > 0$. By Observation 5.3.3, $s_1(t) \geq \rho_2$, $s_2(t) = \rho_2$ and $s_0(t) = 1 - s_1(t)$ if $w_0 > 0$ and $s_0(t) = \min(\rho_0, 1 - s_1(t))$ otherwise. Note that the fluid model keeps $w_2 = 0$ as long as $w_1 > 0$. This results in two extreme directions in which the process $\boldsymbol{w}$ can move. One extreme direction arises from giving priority to class 0, i.e. $s_0(t) = 1 - \rho_2$ and $s_1(t) = \rho_2$. This lets the work of class 0 decrease at rate $1 - \rho_0 - \rho_2 - \mathbf{1}_{(w_0=0)}(1 - \rho_0 - \rho_2)$ and the work of class 1 decrease at rate $\rho_2 - \rho_1 + \mathbf{1}_{(w_0=0)}(1 - \rho_0 - \rho_2)$. The other extreme direction is obtained by giving priority to class 1, so $s_0(t) = 0$ and $s_1(t) = 1$. Then the work of class 1 decreases at rate $1 - \rho_1$ and the work of class 0 increases at rate ρ_0. Figure 5.8 shows the two extreme directions in case $w_0 > 0$. Every direction in the cone between these two directions is also possible. Note that when $\boldsymbol{w} = (w_0, w_1, 0)$, with $w_1 > 0$, every possible direction is work conserving in node 1.
Now we consider the case that $\boldsymbol{w} = (w_0, 0, 0)$. By Observation 5.3.3, $s_1(t) = \rho_1, s_2(t) \geq \rho_1$, so $s_0(t) = 1 - s_2(t)$ if $w_0 > 0$ and $s_0(t) = \min(\rho_0, 1 - s_2(t))$ otherwise. In node 1 the wasted capacity is $s_2(t) - \rho_1$. Hence every policy with $s_2(t) > \rho_1$ is non-work conserving in node 1 at time t. When $s_2(t) = \rho_1$, no capacity is wasted, however then the trajectory certainly leaves the $w_2 = 0$-plane. This results in the following observation.

Observation 5.3.4 *Consider $\boldsymbol{w} = (w_0, w_1, 0)$. As long as $w_1 > 0$, every optimal policy is work conserving in node 1 at time t and will remain in the $w_2 = 0$-plane. As soon as $w_1 = 0$, an optimal policy will be non-work conserving in node 1 or leaves the $w_2 = 0$-plane, or both.*

Assume $\boldsymbol{w}^1, \boldsymbol{w}^2 \in \{\boldsymbol{w} : w_1 \geq 0, w_2 = 0\}$ and denote by $\Pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ the set of all policies that are in Π, and when moving from $\boldsymbol{w}^1$ to $\boldsymbol{w}^2$, are work conserving in node 1 and do not leave the $w_2 = 0$-plane. This set can be empty. The next lemma establishes that minimizing $K^\pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ over all policies $\pi \in \Pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ is achieved by the policy that first assigns priority to class 1, and then to class 0.

Lemma 5.3.5 *Assume $\mu_0 \leq \mu_1$, $\boldsymbol{w}^1 \in \{\boldsymbol{w} : w_1 > 0, w_2 = 0\}$ and $\boldsymbol{w}^2 \in \{\boldsymbol{w} : w_1 \geq 0, w_2 = 0\}$. If $\Pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ is non-empty, then $\text{argmin}_{\pi \in \Pi(\boldsymbol{w}^1, \boldsymbol{w}^2)} K^\pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ is the policy that first assigns priority to class 1, and then to class 0, as depicted in Figure 5.9.*

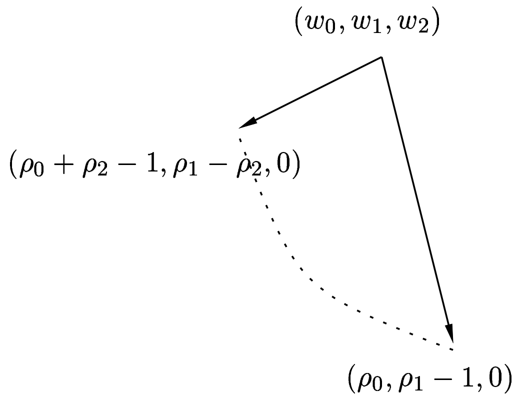

Figure 5.8: Directions when $w_0, w_1 > 0$ and $w_2 = 0$.

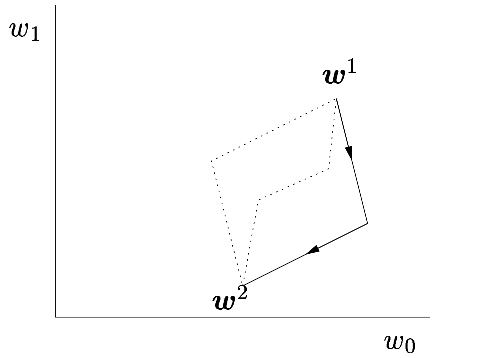

Figure 5.9: Optimal trajectory between $\boldsymbol{w}^1$ and $\boldsymbol{w}^2$ in the $w_2 = 0$-plane.

Proof Every trajectory between $\boldsymbol{w}^1$ and $\boldsymbol{w}^2$ resulting from a policy $\pi \in \Pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ is work conserving in node 1 and keeps the work of class 2 equal to zero. Therefore each trajectory costs the same amount of time and by Observation 5.3.4 it does not hit the horizontal axis, except in $\boldsymbol{w}^2$ when $w_1^2 = 0$. First prioritizing class 1 maximizes in the beginning the departure rate and therefore the policy belonging to this trajectory is the minimizer of $K^\pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$ over $\Pi(\boldsymbol{w}^1, \boldsymbol{w}^2)$. □

We are now able to determine the optimal trajectory. Assume $w_1 > 0$, since otherwise we are in Case 1 for which we already derived the optimal policies. The optimal trajectory is a trajectory that moves from $\boldsymbol{w}$ to $\boldsymbol{0}$. At some point, the optimal trajectory hits the horizontal axis of the $w_2 = 0$-plane for the first time, denote this point by $\bar{\boldsymbol{w}}$. By Observation 5.3.4 we know that the optimal trajectory between $\boldsymbol{w}$ and $\bar{\boldsymbol{w}}$ is work conserving in node 1 and stays in the $w_2 = 0$-plane, which implies that the optimal trajectory is in the set $\Pi(\boldsymbol{w}, \bar{\boldsymbol{w}})$. Applying Lemma 5.3.5 to $\boldsymbol{w}^1 = \boldsymbol{w}$ and $\boldsymbol{w}^2 = \bar{\boldsymbol{w}}$ shows that the shape of the trajectory between $\boldsymbol{w}$ and $\bar{\boldsymbol{w}}$ is as depicted in Figure 5.9. Hence the structure of the average optimal policy up to the time until it hits the horizontal axis is known. Figure 5.10 shows a possible optimal trajectory starting in $\boldsymbol{w}$. From the time that we reach point $\bar{\boldsymbol{w}}$ onward, we will remain in the $w_1 = 0$-plane and we already know what is optimal, depending on whether $\mu_2 \geq \mu_0$ or $\mu_2 \leq \mu_0$.

In order to obtain the average optimal policy, we only have to determine the optimal turning point $\boldsymbol{b}$ as depicted in Figure 5.10. We calculate the costs belonging to the trajectory that turns at $\boldsymbol{b}$. The corresponding policy is denoted by $\pi^{\boldsymbol{b}}$. The time it takes to move from $\boldsymbol{w}$ to $\boldsymbol{b}$ is equal to $T(\boldsymbol{w}, \boldsymbol{b}) = \frac{b_0 - w_0}{\rho_0}$ during which the length of the queue is on average $\frac{w_0 + b_0}{2}\mu_0 + \frac{w_1 + b_1}{2}\mu_1$. The time it takes to move from $\boldsymbol{b}$ to $\bar{\boldsymbol{w}}$ is equal to $T(\boldsymbol{b}, \bar{\boldsymbol{w}}) = \frac{b_1}{\rho_2 - \rho_1}$ during which the length of the queue is on average $\frac{b_0 + \bar{w}_0}{2}\mu_0 + \frac{b_1}{2}\mu_1$, with $\bar{w}_0 = b_0 - \frac{1 - \rho_0 - \rho_2}{\rho_2 - \rho_1} b_1$. The costs under policy $\pi^{\boldsymbol{b}}$ can be written as:

$$K^{\pi^{\boldsymbol{b}}}(\boldsymbol{w}, \boldsymbol{0}) = T(\boldsymbol{w}, \boldsymbol{b})(\frac{w_0 + b_0}{2}\mu_0 + \frac{w_1 + b_1}{2}\mu_1) + T(\boldsymbol{b}, \bar{\boldsymbol{w}})(\frac{b_0 + \bar{w}_0}{2}\mu_0 + \frac{b_1}{2}\mu_1) + K(\bar{\boldsymbol{w}}, \boldsymbol{0}), \quad (5.15)$$

with $b_0 \in [w_0, w_0 + w_1 \frac{\rho_0}{1 - \rho_1}]$ and $b_1 = w_1 - T(\boldsymbol{w}, \boldsymbol{b})(1 - \rho_1)$. This cost function should be minimized over $\boldsymbol{b}$. The term $K(\bar{\boldsymbol{w}}, \boldsymbol{0})$ depends on the value of μ_2 and was already calculated. We consider the two possibilities: $\mu_2 \geq \mu_0$ and $\mu_2 \leq \mu_0$.

Case: $\mu_2 \geq \mu_0$

From (5.13) we know that $K(\bar{\boldsymbol{w}}, \mathbf{0}) = \frac{1}{2}\frac{\bar{w}_0^2}{1-\rho_0-\rho_2}\mu_0$. Plugging this in (5.15) and after some algebraic manipulation, the total costs under policy $\pi^{\boldsymbol{b}}$ can be written as

$$K^{\pi^{\boldsymbol{b}}}(\boldsymbol{w}, \mathbf{0}) = \Big[\frac{1}{2}b_0^2(\mu_1 + \mu_0\frac{\rho_2 - \rho_1}{1-\rho_0-\rho_2}\frac{\rho_0}{1-\rho_1}) - b_0\mu_1(w_0 + \frac{\rho_0}{1-\rho_1}w_1)\Big] \cdot \frac{(1-\rho_1)(1-\rho_2)}{\rho_0^2(\rho_2-\rho_1)} + C,$$

where C is independent of b_0 and $b_0 \in [w_0, w_0 + w_1\frac{\rho_0}{1-\rho_1}]$. Note that $\rho_2 - \rho_1 \geq 0$, therefore this function attains its minimum when b_0 equals

$$b_0^* = \max\Big(w_0, \frac{w_0 + w_1\frac{\rho_0}{1-\rho_1}}{1 + \frac{\mu_0}{\mu_1}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}\frac{\rho_0}{1-\rho_1}}\Big) \leq w_0 + w_1\frac{\rho_0}{1-\rho_1}.$$

It follows that the minimum of the cost function is attained in $b_0^* = w_0$ if and only if $w_1 \leq \frac{\mu_0}{\mu_1}w_0\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}$. The line

$$w_1 = \frac{\mu_0}{\mu_1}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}w_0 \tag{5.16}$$

is now the sought switching curve. When the starting point $\boldsymbol{w}$ is below the switching curve, it is thus optimal to immediately give priority to class 0, since the optimal turning point is $\boldsymbol{b}^* = \boldsymbol{w}$. When $\boldsymbol{w}$ is above the switching curve, the minimum of the cost function is attained in

$$b_0^* = \frac{w_0 + w_1\frac{\rho_0}{1-\rho_1}}{1 + \frac{\mu_0}{\mu_1}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}\frac{\rho_0}{1-\rho_1}}$$

and it may be checked that when starting in a point above the switching curve, the point b_0^* is exactly the point where the trajectory that first gives priority to class 1, hits the switching curve. From then on, class 0 is given preference. The average optimal policy is now completely characterized by a switching curve. In Figure 5.11 the optimal paths for several starting points are drawn. Note that when the trajectory hits the horizontal axis, it continues along this axis, since $\mu_2 \geq \mu_0$ implies that w_1 and w_2 both remain empty, see Figure 5.6.

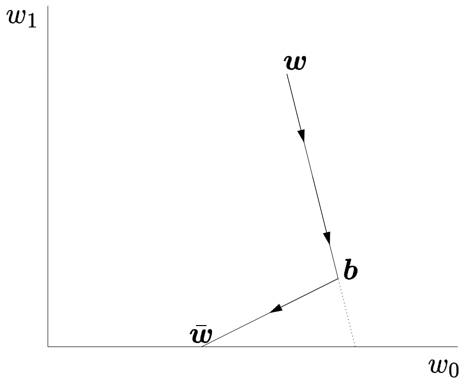

Figure 5.10: Possible optimal trajectory in the $w_2 = 0$-plane when starting in state $\boldsymbol{w}$.

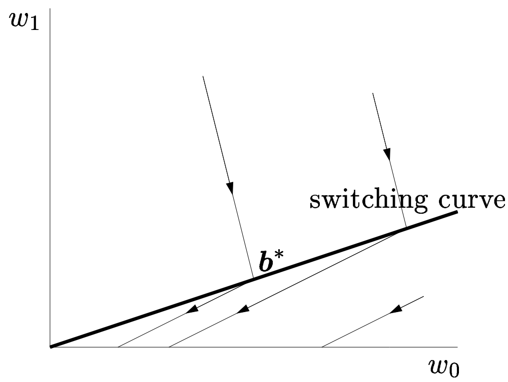

Figure 5.11: Optimal trajectories in the $w_2 = 0$-plane when $w_1 > 0$, if $\mu_0 \leq \mu_1$.

Case: $\mu_2 \le \mu_0$

In this case $K(\bar{\boldsymbol{w}}, \mathbf{0})$ is given by (5.14) and after some algebraic manipulation the cost function in (5.15) can be expressed in b_0 as:

$$\begin{aligned} K(\boldsymbol{w}, \mathbf{0}) = \quad & \Big[\frac{1}{2} b_0^2(\mu_1 + \mu_2 - \mu_0 + \mu_2 \frac{\rho_2 - \rho_1}{1 - \rho_0 - \rho_2} \frac{\rho_0}{1 - \rho_1}) \\ & -b_0(\mu_1 + \mu_2 - \mu_0)(w_0 + \frac{\rho_0}{1 - \rho_1} w_1)\Big] \cdot \frac{(1 - \rho_1)(1 - \rho_2)}{\rho_0^2(\rho_2 - \rho_1)} + C, \end{aligned}$$

with C independent of b_0 and $b_0 \in [w_0, w_0 + w_1 \frac{\rho_0}{1-\rho_1}]$. Note that $\rho_2 - \rho_1 \ge 0$, therefore this function attains its minimum when b_0 equals

$$b_0^* = \max\Big(w_0, \frac{w_0 + w_1 \frac{\rho_0}{1-\rho_1}}{1 + \frac{\mu_2}{\mu_1+\mu_2-\mu_0} \frac{\rho_2-\rho_1}{1-\rho_0-\rho_2} \frac{\rho_0}{1-\rho_1}}\Big) \le w_0 + w_1 \frac{\rho_0}{1 - \rho_1}.$$

It follows that the minimum of the cost function is taken in $b_0^* = w_0$ if and only if $w_1 \le \frac{\mu_2}{\mu_1+\mu_2-\mu_0} w_0 \frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}$. The line

$$w_1 = \frac{\mu_2}{\mu_1 + \mu_2 - \mu_0} \frac{\rho_2 - \rho_1}{1 - \rho_0 - \rho_2} w_0 \tag{5.17}$$

is now the sought switching curve. The average optimal policy is therefore characterized by a switching curve. In Figure 5.11 the optimal trajectories for several starting points are drawn. Note that from the moment that the trajectory hits the horizontal axis on, the optimal trajectory is depicted in Figure 5.7.

5.3.3 Translation back to the original model

In Proposition 5.3.1 optimal policies for the fluid model were derived. In some situations we were able to find a sample-path optimal policy. Such a policy is related to a stochastically optimal policy in the original model. An average optimal policy minimizes $\frac{1}{T}\int_0^T \sum_{i=0}^2 y_i(t)\mathrm{d}t$, hence such a policy is related to a policy that minimizes the average number of users in the original system. In this section we will briefly inspect the optimal policies in the fluid model for various service rate parameters and discuss their performance in the original model.

Case $\mu_0 \ge \mu_1, \mu_2$

In the original model policy π^* is stochastically optimal. This policy serves classes 1 and 2 whenever both are present. In the sample-path optimal policy for the fluid model described in Proposition 5.3.1, we have $s_0 = 0$ when there is work of both classes 1 and 2, so this coincides with π^* in the original model. Further, we have $s_0 = 1 - \rho_1$ or $s_0 = 1 - \rho_2$, when there is no work of class 1 or class 2 present, respectively. Work of each class arrives continuously, so when class i is empty, the optimal policy immediately serves arriving work of class i and the capacity left-over is completely given to class 0. This coincides with policy π^* in the original system, since then class 0 is served, when Q_1 or Q_2 is empty.

Case $\mu_2 \geq \mu_0 \geq \mu_1$

In the fluid model a sample-path optimal policy is found. This policy amounts to giving class 2 preemptive priority over class 0 and if there is no work of class 2, class 0 receives the capacity left-over. In the original model, this would amount to giving class 2 preemptive priority over class 0, and class 1 is only served when class 2 is served or when there is no class-0 work present. This policy is stable under the standard conditions, however it will probably not be a stochastically optimal policy.
In Section 5.2 we proved that in the original system the average optimal policy will always serve classes 1 and 2 whenever both are present and serve class 0 when Q_2 is empty. This part coincides with the fluid solution. However, instead of preemptive priority for class 2, in numerical results we observed a switching curve when Q_1 is empty.

Case $\mu_1 \geq \mu_0 \geq \mu_2$

The average optimal policy found in the fluid model is the policy that allocates no capacity to class 0 when work of both classes 1 and 2 is present, or when the length of the fluid queues is such that $\frac{y_1}{y_0} \geq \frac{\mu_2}{\mu_1+\mu_2-\mu_0}\frac{\mu_1}{\mu_0}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}$. Otherwise class 0 receives exactly the fraction that is left-over to keep either class 1 or class 2 empty. This policy translated to the original model is a policy that serves classes 1 and 2 whenever users of both classes are present and which serves class 0 when only users of classes 0 and 2 are present. Note that these are exactly the properties an average optimal policy should have, according to Lemmas 5.1 and 5.2. Further, the policy has a switching curve when only classes 0 and 1 are present. This coincides with the switching curve structure of the optimal policies found in Section 5.1.

In the fluid model, the slope of the switching curve is equal to $\frac{\mu_2}{\mu_1+\mu_2-\mu_0}\frac{\mu_1}{\mu_0}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}$. When we calculate this expression for the three parameter sets from Table 5.1, we get $\frac{14}{15}$, $\frac{7}{15}$ and $\frac{8}{10}$, respectively. This matches with the slopes observed in Figures 5.1–5.3, hence the slopes of the switching curves correspond as well.

Also the effects observed in Section 5.1 in the switching curve when changing the time scale is reflected in the analytic expression for the switching curve in the fluid model. Changing the time scale means that the traffic load ρ_i remains constant, while the service rate μ_i is changed (and proportionally the arrival rate λ_i). The slope is equal to $\frac{\mu_2}{\mu_1+\mu_2-\mu_0}\frac{\mu_1}{\mu_0}\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}$. From $\mu_1 \geq \mu_0$, it follows that the slope is increasing in μ_2 and from $\mu_2 \leq \mu_0$ it follows that the slope is decreasing in μ_1. This is exactly what we observed in the numerical experiments.

Case $\mu_1, \mu_2 \geq \mu_0$

The average optimal policy found in the fluid model is the policy that allocates no capacity to class 0 when work of class 2 is present, or when the length of the fluid queue is such that $\frac{y_1}{y_0} \geq \frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}$. Otherwise class 0 receives $1-\rho_2$. In the original model, this amounts to serving class 2 when Q_2 is non-empty and a switching curve decides which class to serve when Q_2 is empty. When both Q_1 and Q_2 are non-empty, according to Lemma 5.1 it is indeed optimal in the original model to serve classes 1 and 2. When Q_2 is empty, in Section 5.1 a switching curve structure was found as well.

Consider the set of parameters belonging to Figures 5.4 and 5.5. We first discuss the optimal actions when Q_2 is empty. The slope of the switching curve for the fluid model is equal to $\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}$, which equals 3/4 for this set of parameters. This matches with the slope observed in Figure 5.4. Further numerical experiments showed that in the original model changing the time scales had no influence on the slope. This corresponds with the fluid model, since the slope $\frac{\rho_2-\rho_1}{1-\rho_0-\rho_2}$ only depends on the traffic loads. Now consider the optimal actions when Q_1 is empty. In Figure 5.5 we observe that it is almost always optimal to serve class 2, as suggested by the fluid solution as well.

Further, note that if $\rho_1 = \rho_2$, this policy amounts to giving preemptive priority to classes 1 and 2 and this will certainly not be the optimal policy in the original model, since we know that this may cause unnecessarily instability. From the solution of the fluid model, we can only conclude that the switching curves for the original model, when class 1 respectively class 2 is empty, are not that significant, since they do not arise in the fluid model. The reason why these switching curves do not appear in the solution for the fluid model, is that the latter is completely deterministic and involves no stochasticity. Hence, it is guaranteed that work of class i arrives at a continuous rate of ρ_i. In the original model we do not have this certainty, which is why switching curves manifest themselves.

Bibliography

[1] Avram, F. (1997). Optimal control of fluid limits of queueing networks and stochasticity corrections. *Lectures in Applied Mathematics* **33**, 1–36.

[2] Bonald, T., Massoulié, L. (2001). Impact of fairness on Internet performance. In: *Proc. ACM Sigmetrics & Performance 2001 Conf.*, Boston MA, USA, 82–91.

[3] Koole, G.M. (1992). *Stochastic scheduling and dynamic programming.* PhD thesis.

[4] López, F.J., Sanz, G. (2002). Markovian couplings staying in arbitrary subsets of the state space. *Journal of Applied Probability* **39**, 197–212.

[5] Mo, J., Walrand, J. (2000). Fair end-to-end window-based congestion control. *IEEE/ACM Transactions on Networking* **8**, 556–567.

[6] Reich, E. (1958). On the integrodifferential equation of Takács I. *Annals of Mathematical Statistics* **29**, 563–570.

[7] Righter, R., Shanthikumar, J.G. (1989). Scheduling multiclass single server queueing systems to stochastically maximize the number of successful departures. *Probability in the Engineering and Informational Sciences* **3**, 323–333.

[8] Schrage, L.E. (1968). A proof of the optimality of the shortest remaining processing time discipline. *Operations Research* **16**, 687–690.

[9] Schrage, L.E., Miller, L.W. (1966). The queue M/G/1 with the shortest remaining processing time discipline. *Operations Research* **14**, 670–684.

[10] Shaked, M., Shanthikumar, J.G. (1994). *Stochastic orders and their applications.* Academic Press.

[11] Tijms, H.C. (2003). *A first course in stochastic models.* Wiley.

[12] Verloop, I.M., Borst, S.C., Núñez Queija, R. (2005). *Stability of size-based scheduling disciplines in resource-sharing networks.* To appear in: *Proc. Performance 2005.*

[13] Weiss, G. (1995). *Optimal draining of a fluid re-entrant line. IMA volumes in Mathematics and its Applications* **71**, 91–103.

[14] Yang, S., De Veciana, G. (2002). Size-based adaptive bandwidth allocation: optimizing the average QoS for elastic flows. In: *Proc. IEEE Infocom 2002*, New York NY, USA, 238–249.

[15] Yang, S., De Veciana, G. (2004). Enhancing both network and user performance for networks supporting best-effort traffic. *IEEE/ACM Transactions on Networking* **12**, 349–360.

[16] Ye, H.Q. (2003). Stability of data networks under an optimization-based bandwidth allocation. *IEEE Transactions on Automatic Control* **48**, 1238–1242.

Appendix A

Proof of Lemma 5.2.3

In this section we will prove Lemma 5.2.3 by induction on n. For $W = V_0$ it holds. Suppose we know that the seven inequalities (5.6)–(5.12) hold for $W = V_n$. We now show that they hold for $W = V_{n+1}$ as well. Define

$$\begin{aligned}\tilde{V}_{n+1}(i,j,k) &= V_{n+1}(i,j,k) - c(i,j,k) - \lambda_0 V_n(i+1,j,k) - \lambda_1 V_n(i,j+1,k) - \lambda_2 V_n(i,j,k+1) \\ &= \min\{\mu_0 V_n((i-1)^+,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k), \\ &\qquad \mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,(k-1)^+)\}.\end{aligned}$$

Since the seven inequalities hold for both c and V_n, it is straightforward to check that $c(i,j,k) + \lambda_0 V_n(i+1,j,k) + \lambda_1 V_n(i,j+1,k) + \lambda_2 V_n(i,j,k+1)$ satisfies these inequalities as well. In order to prove that V_{n+1} satisfies (5.6) – (5.12), it is therefore sufficient to show that $\tilde{V}_{n+1}$ does.
The following observation can be made, which will be helpful in proving the seven inequalities. The seven inequalities being true for $W = V_n$, implicitly means that at time $n+1$ the optimal actions are of a switching curve structure. So for example, if at time $n+1$ it is optimal to serve class 1 when we are in state (i,j,k), this is also optimal if at time $n+1$ we are in state $(i,j+1,k)$. This will be referred to as $\tilde{V}_{n+1}$ following a switching curve.

Proof of inequality (5.6)
We have to show that $W = \tilde{V}_{n+1}$ satisfies (5.6), that is

$$\begin{aligned}&\mu_0 \tilde{V}_{n+1}(i,j+1,k) + \mu_0 \tilde{V}_{n+1}(i-1,j,k) + 2\mu_1 \tilde{V}_{n+1}(i,j,k) \\ &\le \mu_0 \tilde{V}_{n+1}(i,j,k) + \mu_0 \tilde{V}_{n+1}(i-1,j+1,k) + \mu_1 \tilde{V}_{n+1}(i,(j-1)^+,k) + \mu_1 \tilde{V}_{n+1}(i,j+1,k). \quad \text{(A.1)}\end{aligned}$$

The value of the RHS of (A.1) depends on which actions are optimal in the states $(i,j,k), (i-1,j+1,k), (i,(j-1)^+,k)$ and $(i,j+1,k)$. In every state there are two possibilities, either serve classes 1 and 2, or serve class 0. Since $\tilde{V}_{n+1}$ follows a switching curve, only five possible combinations of optimal actions in the various states can occur. This corresponds to the following five situations. In situation 1 it is optimal to serve classes 1 and 2 in states $(i,j,k), (i-1,j+1,k)$ and $(i,j+1,k)$ and class 0 in state $(i,(j-1)^+,k)$. In situation 2 it is optimal to serve classes 1 and 2 in states $(i-1,j+1,k)$ and $(i,j+1,k)$ and class 0 in states (i,j,k) and $(i,j+1,k)$. In situation 3 it is optimal to serve classes 1 and 2 in state $(i-1,j+1,k)$ and serve class 0 in the other three states. In situation 4 it is optimal to serve class 0 in all four states and in situation 5 it is optimal to serve classes 1 and 2 in all four states. For the first three situations, the RHS of (A.1) is written out and this yields:

$$\begin{aligned}
\text{Situation 1:}\quad & \mu_0[\mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,(k-1)^+)] \\
& +\mu_0[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i-1,j,k) + \mu_2 V_n(i-1,j+1,(k-1)^+)] \\
& +\mu_1[\mu_0 V_n(i-1,(j-1)^+,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,(j-1)^+,k)] \\
& +\mu_1[\mu_0 V_n(i,j+1,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j+1,(k-1)^+)], \\
\text{Situation 2:}\quad & \mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k)] \\
& +\mu_0[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i-1,j,k) + \mu_2 V_n(i-1,j+1,(k-1)^+)] \\
& +\mu_1[\mu_0 V_n(i-1,(j-1)^+,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,(j-1)^+,k)] \\
& +\mu_1[\mu_0 V_n(i,j+1,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j+1,(k-1)^+)], \\
\text{Situation 3:}\quad & \mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k)] \\
& +\mu_0[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i-1,j,k) + \mu_2 V_n(i-1,j+1,(k-1)^+)] \\
& +\mu_1[\mu_0 V_n(i-1,(j-1)^+,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,(j-1)^+,k)] \\
& +\mu_1[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i,j+1,k) + \mu_2 V_n(i,j+1,k)].
\end{aligned}$$

For each of the five possible situations we will show that (A.1) holds.

- Situation 1: We can write

$$\begin{aligned}
& \mu_0 \tilde{V}_{n+1}(i,j+1,k) + \mu_0 \tilde{V}_{n+1}(i-1,j,k) + 2\mu_1 \tilde{V}_{n+1}(i,j,k) \\
& \leq \mu_0[\mu_0 V_n(i,j+1,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j+1,(k-1)^+)] \\
& \quad +\mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i-1,(j-1)^+,k) + \mu_2 V_n(i-1,j,(k-1)^+)] \\
& \quad +\mu_1[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k)] \\
& \quad +\mu_1[\mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,(k-1)^+)]. \qquad \text{(A.2)}
\end{aligned}$$

The terms in (A.2) with a factor μ_2 are

$$\begin{aligned}
& \mu_2[\mu_0 V_n(i,j+1,(k-1)^+) + \mu_0 V_n(i-1,j,(k-1)^+) + \mu_1 V_n(i,j,k) + \mu_1 V_n(i,j,(k-1)^+)] \\
& \leq \mu_2[\mu_0 V_n(i,j,(k-1)^+) + \mu_0 V_n(i-1,j+1,(k-1)^+) + \mu_1 V_n(i,(j-1)^+,(k-1)^+) \\
& \quad +\mu_1 V_n(i,j+1,(k-1)^+) - \mu_1 V_n(i,j,(k-1)^+) + \mu_1 V_n(i,j,k)] \\
& \leq \mu_2[\mu_0 V_n(i,j,(k-1)^+) + \mu_0 V_n(i-1,j+1,(k-1)^+) \\
& \quad +\mu_1 V_n(i,(j-1)^+,k) + \mu_1 V_n(i,j+1,(k-1)^+)],
\end{aligned}$$

where the first inequality follows from (5.6) and the second from (5.12). By (5.6) the remaining terms in (A.2) are smaller than or equal to

$$\begin{aligned}
& \mu_0[\mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k)] \\
& +\mu_0[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i-1,j,k)] \\
& +\mu_1[\mu_0 V_n(i-1,(j-1)^+,k) + \mu_1 V_n(i,(j-1)^+,k)] \\
& +\mu_1[\mu_0 V_n(i,j+1,k) + \mu_1 V_n(i,j,k)].
\end{aligned}$$

Together this gives what was to be proved, namely that

$$\begin{aligned}
&\mu_0\tilde{V}_{n+1}(i,j+1,k)+\mu_0\tilde{V}_{n+1}(i-1,j,k)+2\mu_1\tilde{V}_{n+1}(i,j,k)\\
&\leq \mu_0[\mu_0V_n(i,j,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_2V_n(i,j,(k-1)^+)]\\
&\quad+\mu_0[\mu_0V_n(i-1,j+1,k)+\mu_1V_n(i-1,j,k)+\mu_2V_n(i-1,j+1,(k-1)^+)]\\
&\quad+\mu_1[\mu_0V_n(i-1,(j-1)^+,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_2V_n(i,(j-1)^+,k)]\\
&\quad+\mu_1[\mu_0V_n(i,j+1,k)+\mu_1V_n(i,j,k)+\mu_2V_n(i,j+1,(k-1)^+)]\\
&=\mu_0\tilde{V}_{n+1}(i,j,k)+\mu_0\tilde{V}_{n+1}(i-1,j+1,k)+\mu_1\tilde{V}_{n+1}(i,(j-1)^+,k)+\mu_1\tilde{V}_{n+1}(i,j+1,k).
\end{aligned}$$

- Situation 2: We can write

$$\begin{aligned}
&\mu_0\tilde{V}_{n+1}(i,j+1,k)+\mu_0\tilde{V}_{n+1}(i-1,j,k)+2\mu_1\tilde{V}_{n+1}(i,j,k)\\
&\leq \mu_0[\mu_0V_n(i-1,j+1,k)+\mu_1V_n(i,j+1,k)+\mu_2V_n(i,j+1,k)]\\
&\quad+\mu_0[\mu_0V_n(i-1,j,k)+\mu_1V_n(i-1,(j-1)^+,k)+\mu_2V_n(i-1,j,(k-1)^+)]\\
&\quad+\mu_1[\mu_0V_n(i-1,j,k)+\mu_1V_n(i,j,k)+\mu_2V_n(i,j,k)]\\
&\quad+\mu_1[\mu_0V_n(i,j,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_2V_n(i,j,(k-1)^+)].
\end{aligned} \tag{A.3}$$

The terms in (A.3) with a factor μ_2 are

$$\begin{aligned}
&\mu_2[\mu_0V_n(i,j+1,k)+\mu_0V_n(i-1,j,(k-1)^+)+\mu_1V_n(i,j,k)+\mu_1V_n(i,j,(k-1)^+)]\\
&\leq \mu_2[\mu_0V_n(i-1,j,(k-1)^+)-\mu_0V_n(i-1,j,k)+\mu_1V_n(i,j,(k-1)^+)-\mu_1V_n(i,j,k)\\
&\quad+\mu_0V_n(i,j,k)+\mu_0V_n(i-1,j+1,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_1V_n(i,j+1,k)].\\
&\leq \mu_2[\mu_0V_n(i,j,k)+\mu_0V_n(i-1,j+1,(k-1)^+)\\
&\quad+\mu_1V_n(i,(j-1)^+,k)+\mu_1V_n(i,j+1,(k-1)^+)],
\end{aligned}$$

where the first inequality follows from (5.6) and the second from (5.12). This inequality together with (A.3) gives what was to be proved, namely that

$$\begin{aligned}
&\mu_0\tilde{V}_{n+1}(i,j+1,k)+\mu_0\tilde{V}_{n+1}(i-1,j,k)+2\mu_1\tilde{V}_{n+1}(i,j,k)\\
&\leq \mu_0[\mu_0V_n(i-1,j,k)+\mu_1V_n(i,j,k)+\mu_2V_n(i,j,k)]\\
&\quad+\mu_0[\mu_0V_n(i-1,j+1,k)+\mu_1V_n(i-1,j,k)+\mu_2V_n(i-1,j+1,(k-1)^+)]\\
&\quad+\mu_1[\mu_0V_n(i-1,(j-1)^+,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_2V_n(i,(j-1)^+,k)]\\
&\quad+\mu_1[\mu_0V_n(i,j+1,k)+\mu_1V_n(i,j,k)+\mu_2V_n(i,j+1,(k-1)^+)]\\
&=\mu_0\tilde{V}_{n+1}(i,j,k)+\mu_0\tilde{V}_{n+1}(i-1,j+1,k)+\mu_1\tilde{V}_{n+1}(i,(j-1)^+,k)+\mu_1\tilde{V}_{n+1}(i,j+1,k).
\end{aligned}$$

- Situation 3: We can write

$$\begin{aligned}
&\mu_0\tilde{V}_{n+1}(i,j+1,k)+\mu_0\tilde{V}_{n+1}(i-1,j,k)+2\mu_1\tilde{V}_{n+1}(i,j,k)\\
&\leq \mu_0[\mu_0V_n(i-1,j+1,k)+\mu_1V_n(i,j+1,k)+\mu_2V_n(i,j+1,k)]\\
&\quad+\mu_0[\mu_0V_n(i-1,j,k)+\mu_1V_n(i-1,(j-1)^+,k)+\mu_2V_n(i-1,j,(k-1)^+)]\\
&\quad+\mu_1[\mu_0V_n(i-1,j,k)+\mu_1V_n(i,j,k)+\mu_2V_n(i,j,k)]\\
&\quad+\mu_1[\mu_0V_n(i-1,j,k)+\mu_1V_n(i,j,k)+\mu_2V_n(i,j,k)].
\end{aligned} \tag{A.4}$$

The terms in (A.4) with a factor μ_2 are

$$\begin{aligned}
&\mu_2[\mu_0 V_n(i,j+1,k) + \mu_0 V_n(i-1,j,(k-1)^+) + \mu_1 V_n(i,j,k) + \mu_1 V_n(i,j,k)] \\
&\leq \mu_2[\mu_0 V_n(i,j,k) + \mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_1 V_n(i,j+1,k) \\
&\quad -\mu_0 V_n(i-1,j,(k-1)^+) - \mu_0 V_n(i-1,j,k)] \\
&\leq \mu_2[\mu_0 V_n(i,j,k) + \mu_0 V_n(i-1,j+1,(k-1)^+) + \mu_1 V_n(i,(j-1)^+,k) + \mu_1 V_n(i,j+1,k)],
\end{aligned}$$

where the first inequality follows from (5.6) and the second from (5.12). By (5.6) the remaining terms in (A.4) are smaller than or equal to

$$\begin{aligned}
&\mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k)] \\
&+\mu_0[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i-1,j,k)] \\
&+\mu_1[\mu_0 V_n(i-1,(j-1)^+,k) + \mu_1 V_n(i,(j-1)^+,k)] \\
&+\mu_1[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i,j+1,k)].
\end{aligned}$$

Together this gives what was to be proved, namely that

$$\begin{aligned}
&\mu_0 \tilde{V}_{n+1}(i,j+1,k) + \mu_0 \tilde{V}_{n+1}(i-1,j,k) + 2\mu_1 \tilde{V}_{n+1}(i,j,k) \\
&\leq \mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k)] \\
&\quad +\mu_0[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i-1,j,k) + \mu_2 V_n(i-1,j+1,(k-1)^+)] \\
&\quad +\mu_1[\mu_0 V_n(i-1,(j-1)^+,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,(j-1)^+,k)] \\
&\quad +\mu_1[\mu_0 V_n(i-1,j+1,k) + \mu_1 V_n(i,j+1,k) + \mu_2 V_n(i,j+1,k)] \\
&= \mu_0 \tilde{V}_{n+1}(i,j,k) + \mu_0 \tilde{V}_{n+1}(i-1,j+1,k) + \mu_1 \tilde{V}_{n+1}(i,(j-1)^+,k) + \mu_1 \tilde{V}_{n+1}(i,j+1,k).
\end{aligned}$$

- Situation 4: We can only be in this situation if $i > 1$, since V_n is increasing in i, j and k. The inequality then easily follows from (5.8).

- Situation 5: The inequality easily follows from (5.8).

Proof of inequality (5.7)
Inequality (5.7) can be proved in the same way as inequality (5.6).

Proof of inequality (5.8)
We have to show that $W = \tilde{V}_{n+1}$ satisfies (5.8), that is

$$\begin{aligned}
&2\mu_0 \tilde{V}_{n+1}(i,j,k) + \mu_1 \tilde{V}_{n+1}(i+1,j,k) + \mu_1 \tilde{V}_{n+1}(i,(j-1)^+,k) \\
&\leq \mu_0 \tilde{V}_{n+1}(i+1,j,k) + \mu_0 \tilde{V}_{n+1}(i-1,j,k) + \mu_1 \tilde{V}_{n+1}(i+1,(j-1)^+,k) + \mu_1 \tilde{V}_{n+1}(i,j,k). \quad \text{(A.5)}
\end{aligned}$$

The value of the RHS of (A.5), depends on which actions are optimal in the four states $(i+1,j,k), (i-1,j,k), (i+1,(j-1)^+,k)$ and (i,j,k). Since the optimal actions at time $n+1$ have a switching structure (as explained earlier), there are exactly 5 possible situations. For the first three situations, the RHS of (A.5) is written out and this yields:

$$\begin{aligned}
\text{Situation 1:} \quad & \mu_0[\mu_0 V_n(i+1,j,k) + \mu_1 V_n(i+1,(j-1)^+,k) + \mu_2 V_n(i+1,j,(k-1)^+)] \\
& +\mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i-1,(j-1)^+,k) + \mu_2 V_n(i-1,j,(k-1)^+)] \\
& +\mu_1[\mu_0 V_n(i,(j-1)^+,k) + \mu_1 V_n(i+1,(j-1)^+,k) + \mu_2 V_n(i+1,(j-1)^+,k)] \\
& +\mu_1[\mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,(k-1)^+)], \qquad \text{(A.6)} \\
\text{Situation 2:} \quad & \mu_0[\mu_0 V_n(i,j,k) + \mu_1 V_n(i+1,j,k) + \mu_2 V_n(i+1,j,k)] \\
& +\mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i-1,(j-1)^+,k) + \mu_2 V_n(i-1,j,(k-1)^+)] \\
& +\mu_1[\mu_0 V_n(i,(j-1)^+,k) + \mu_1 V_n(i+1,(j-1)^+,k) + \mu_2 V_n(i+1,(j-1)^+,k)] \\
& +\mu_1[\mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,(k-1)^+)], \qquad \text{(A.7)} \\
\text{Situation 3:} \quad & \mu_0[\mu_0 V_n(i,j,k) + \mu_1 V_n(i+1,j,k) + \mu_2 V_n(i+1,j,k)] \\
& +\mu_0[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i-1,(j-1)^+,k) + \mu_2 V_n(i-1,j,(k-1)^+)] \\
& +\mu_1[\mu_0 V_n(i,(j-1)^+,k) + \mu_1 V_n(i+1,(j-1)^+,k) + \mu_2 V_n(i+1,(j-1)^+,k)] \\
& +\mu_1[\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k)]. \qquad \text{(A.8)}
\end{aligned}$$

Situations 4 and 5 correspond to either serving class 0 in all four states or serving classes 1 and 2 in all four states. For each of the five possible situations we will show that (A.5) holds.

- Situation 1: We can write

$$\begin{aligned}
& 2\mu_0 \tilde{V}_{n+1}(i,j,k) + \mu_1 \tilde{V}_{n+1}(i+1,j,k) + \mu_1 \tilde{V}_{n+1}(i,(j-1)^+,k) \\
& \leq 2\mu_0[\mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,(k-1)^+)] \\
& \quad +\mu_1[\mu_0 V_n(i+1,j,k) + \mu_1 V_n(i+1,(j-1)^+,k) + \mu_2 V_n(i+1,j,(k-1)^+)] \\
& \quad +\mu_1[\mu_0 V_n(i-1,(j-1)^+,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,(j-1)^+,k)]. \qquad \text{(A.9)}
\end{aligned}$$

The terms in (A.9) with a factor μ_2 are

$$\begin{aligned}
& \mu_2[2\mu_0 V_n(i,j,(k-1)^+) + \mu_1 V_n(i+1,j,(k-1)^+) + \mu_1 V_n(i,(j-1)^+,k)] \\
& \leq \mu_2[\mu_0 V_n(i+1,j,(k-1)^+) + \mu_0 V_n(i-1,j,(k-1)^+) + \mu_1 V_n(i+1,(j-1)^+,(k-1)^+) \\
& \quad +\mu_1 V_n(i,j,(k-1)^+) - \mu_1 V_n(i,(j-1)^+,(k-1)^+) + \mu_1 V_n(i,(j-1)^+,k)] \\
& \leq \mu_2[\mu_0 V_n(i+1,j,(k-1)^+) + \mu_0 V_n(i-1,j,(k-1)^+) \\
& \quad +\mu_1 V_n(i+1,(j-1)^+,k) + \mu_1 V_n(i,j,(k-1)^+)],
\end{aligned}$$

where the first inequality follows from (5.8) and the second from (5.10) (when taking state $(i+1,(j-1)^+,k)$ instead of (i,j,k)). By (5.8), we have

$$\begin{aligned}
& \mu_0[2\mu_0 V_n(i,j,k) + \mu_1 V_n(i+1,j,k) + \mu_1 V_n(i,(j-1)^+,k)] \\
& \leq \mu_0[\mu_0 V_n(i+1,j,k) + \mu_0 V_n(i-1,j,k) + \mu_1 V_n(i+1,(j-1)^+,k) + \mu_1 V_n(i,j,k)].
\end{aligned}$$

Together this gives that (A.9) is not larger than (A.6), which was to be proved.

- Situation 2: We can write

$$\begin{aligned}
&2\mu_0\tilde{V}_{n+1}(i,j,k)+\mu_1\tilde{V}_{n+1}(i+1,j,k)+\mu_1\tilde{V}_{n+1}(i,(j-1)^+,k)\\
&\leq \mu_0[\mu_0V_n(i-1,j,k)+\mu_1V_n(i,j,k)+\mu_2V_n(i,j,k)]\\
&\quad+\mu_0[\mu_0V_n(i,j,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_2V_n(i,j,(k-1)^+)]\\
&\quad+\mu_1[\mu_0V_n(i+1,j,k)+\mu_1V_n(i+1,(j-1)^+,k)+\mu_2V_n(i+1,j,(k-1)^+)]\\
&\quad+\mu_1[\mu_0V_n(i-1,(j-1)^+,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_2V_n(i,(j-1)^+,k)]. \quad (A.10)
\end{aligned}$$

The terms in (A.10) with a factor μ_2 are

$$\begin{aligned}
&\mu_2[\mu_0V_n(i,j,k)+\mu_0V_n(i,j,(k-1)^+)+\mu_1V_n(i+1,j,(k-1)^+)+\mu_1V_n(i,(j-1)^+,k)]\\
&\leq \mu_2[\mu_0V_n(i,j,(k-1)^+)-\mu_0V_n(i,j,k)+\mu_1V_n(i+1,j,(k-1)^+)-\mu_1V_n(i+1,j,k)\\
&\quad+\mu_0V_n(i+1,j,k)+\mu_0V_n(i-1,j,k)+\mu_1V_n(i+1,(j-1)^+,k)+\mu_1V_n(i,j,k)]\\
&\leq \mu_2[\mu_0V_n(i+1,j,k)+\mu_0V_n(i-1,j,(k-1)^+)\\
&\quad+\mu_1V_n(i+1,(j-1)^+,k)+\mu_1V_n(i,j,(k-1)^+)],
\end{aligned}$$

in which the first inequality follows from (5.8) and the second from (5.10). We can conclude that (A.10) is not larger than (A.7), which was to be proved.

- Situation 3: We can write

$$\begin{aligned}
&2\mu_0\tilde{V}_{n+1}(i,j,k)+\mu_1\tilde{V}_{n+1}(i+1,j,k)+\mu_1\tilde{V}_{n+1}(i,(j-1)^+,k)\\
&\leq \mu_0[\mu_0V_n(i-1,j,k)+\mu_1V_n(i,j,k)+\mu_2V_n(i,j,k)]\\
&\quad+\mu_0[\mu_0V_n(i,j,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_2V_n(i,j,(k-1)^+)]\\
&\quad+\mu_1[\mu_0V_n(i,j,k)+\mu_1V_n(i+1,j,k)+\mu_2V_n(i+1,j,k)]\\
&\quad+\mu_1[\mu_0V_n(i-1,(j-1)^+,k)+\mu_1V_n(i,(j-1)^+,k)+\mu_2V_n(i,(j-1)^+,k)]. \quad (A.11)
\end{aligned}$$

The terms in (A.11) with a factor μ_2 are

$$\begin{aligned}
&\mu_2[\mu_0V_n(i,j,k)+\mu_0V_n(i,j,(k-1)^+)+\mu_1V_n(i+1,j,k)+\mu_1V_n(i,(j-1)^+,k)]\\
&\leq \mu_2[\mu_0V_n(i+1,j,k)+\mu_0V_n(i-1,j,k)+\mu_1V_n(i+1,(j-1)^+,k)+\mu_1V_n(i,j,k)\\
&\quad-\mu_0V_n(i,j,k)+\mu_0V_n(i,j,(k-1)^+)]\\
&\leq \mu_2[\mu_0V_n(i+1,j,k)+\mu_0V_n(i-1,j,(k-1)^+)+\mu_1V_n(i+1,(j-1)^+,k)+\mu_1V_n(i,j,k)],
\end{aligned}$$

in which the first inequality follows from (5.8) and the second from (5.10). By (5.8), the remaining terms in (A.11) are smaller than or equal to

$$\begin{aligned}
&\mu_0[\mu_0V_n(i,j,k)+\mu_1V_n(i+1,j,k)]\\
&+\mu_0[\mu_0V_n(i-1,j,k)+\mu_1V_n(i-1,(j-1)^+,k)]\\
&+\mu_1[\mu_0V_n(i,(j-1)^+,k)+\mu_1V_n(i+1,(j-1)^+,k)]\\
&+\mu_1[\mu_0V_n(i-1,j,k)+\mu_1V_n(i,j,k)].
\end{aligned}$$

Together this gives that (A.11) is not larger than (A.8), which was to be proved.

- Situation 4: We can only be in this situation if $i > 1$. The inequality then easily follows from (5.8).

- Situation 5: The inequality easily follows from (5.8).

Proof of inequality (5.9)
Inequality (5.9) can be proved in the same way as (5.8).

Proof of inequality (5.10)
We have to show that $W = \tilde{V}_{n+1}$ satisfies (5.10), that is

$$\tilde{V}_{n+1}(i,j,(k-1)^+) + \tilde{V}_{n+1}((i-1)^+,j,k) \leq \tilde{V}_{n+1}(i,j,k) + \tilde{V}_{n+1}((i-1)^+,j,(k-1)^+). \quad \text{(A.12)}$$

For $i = 0$ or $k = 0$ this is trivially true, so from now on we assume that $i > 0$ and $k > 0$. The value of the RHS of (A.12) depends on which of the two actions are optimal in the two states (i,j,k) and $(i-1,j,k-1)$. This means that there are four situations to be considered in which the RHS of (A.12) has the following values:

Situation 1:
$$\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k) + \mu_0 V_n((i-2)^+,j,k-1) + \mu_1 V_n(i-1,j,k-1) + \mu_2 V_n(i-1,j,k-1),$$

Situation 2:
$$\mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,k-1) + \mu_0 V_n(i-1,j,k-1) + \mu_1 V_n(i-1,(j-1)^+,k-1) + \mu_2 V_n(i-1,j,(k-2)^+),$$

Situation 3:
$$\mu_0 V_n(i,j,k) + \mu_1 V_n(i,(j-1)^+,k) + \mu_2 V_n(i,j,k-1) + \mu_0 V_n((i-2)^+,j,k-1) + \mu_1 V_n(i-1,j,k-1) + \mu_2 V_n(i-1,j,k-1), \quad \text{(A.13)}$$

Situation 4:
$$\mu_0 V_n(i-1,j,k) + \mu_1 V_n(i,j,k) + \mu_2 V_n(i,j,k) + \mu_0 V_n(i-1,j,k-1) + \mu_1 V_n(i-1,(j-1)^+,k-1) + \mu_2 V_n(i-1,j,(k-2)^+). \quad \text{(A.14)}$$

For each of the four possible situations we will show that (A.12) holds.

- Situations 1&2: The inequality follows from (5.10).

We can write

$$\begin{aligned}
&\tilde{V}_{n+1}(i,j,k-1) + \tilde{V}_{n+1}(i-1,j,k) \\
&\leq \mu_0 V_n(i-1,j,k-1) + \mu_1 V_n(i,j,k-1) + \mu_2 V_n(i,j,k-1) \\
&\quad + \mu_0 V_n(i-1,j,k) + \mu_1 V_n(i-1,(j-1)^+,k) + \mu_2 V_n(i-1,j,k-1). \qquad \text{(A.15)}
\end{aligned}$$

For both situations 3 and 4 we will show that the RHS in (A.15) is not larger than (A.13) and (A.14), respectively, which then implies inequality (A.12).

- Situation 3: Situation 3 implies that in state $(i-1,j,k-1)$ serving class 0 is the minimizer. Since V_n is increasing, this can never be the case if $i = 1$. So we may assume that $i > 1$. By (5.8), the terms in (A.15) with factors μ_0 and μ_1 are not larger than

$$\begin{aligned}
&-\mu_0 V_n(i-1,j,k) + \mu_0 V_n(i-1,j,k-1) + \mu_1 V_n(i,j,k-1) - \mu_1 V_n(i,j,k) \\
&+\mu_0 V_n(i,j,k) + \mu_0 V_n(i-2,j,k) + \mu_1 V_n(i,j-1,k) + \mu_1 V_n(i-1,j,k) \\
&= \mu_0 V_n(i,j,k) + \mu_1 V_n(i,j-1,k) + \mu_0 V_n(i-2,j,k-1) + \mu_1 V_n(i-1,j,k-1) \\
&\quad +\mu_0 V_n(i-2,j,k) + \mu_0 V_n(i-1,j,k-1) - \mu_0 V_n(i-2,j,k-1) - \mu_0 V_n(i-1,j,k) \\
&\quad +\mu_1 V_n(i,j,k-1) + \mu_1 V_n(i-1,j,k) - \mu_1 V_n(i,j,k) - \mu_1 V_n(i-1,j,k-1) \\
&\leq \mu_0 V_n(i,j,k) + \mu_1 V_n(i,j-1,k) + \mu_0 V_n(i-2,j,k-1) + \mu_1 V_n(i-1,j,k-1),
\end{aligned}$$

where the inequality follows from (5.10), since $\mu_0 V_n(i-2,j,k) + \mu_0 V_n(i-1,j,k-1) \leq \mu_0 V_n(i-2,j,k-1) + \mu_0 V_n(i-1,j,k)$ and $\mu_1 V_n(i,j,k-1) + \mu_1 V_n(i-1,j,k) \leq \mu_1 V_n(i,j,k) + \mu_1 V_n(i-1,j,k-1)$. Together this gives that (A.15) is not larger than (A.13), which was to be proved.

- Situation 4: From (5.7) and (5.10) we can conclude

$$2V_n(i,j,k) \leq V_n(i,j,(k-1)^+) + V_n(i,j,k+1), \quad \text{for } \; i>0, j \geq 0, k \geq 0. \tag{A.16}$$

The terms with a factor μ_2 in (A.15) are

$$\begin{aligned} &\mu_2[V_n(i,j,k-1) + V_n(i-1,j,k-1)] \\ &\leq \mu_2[V_n(i-1,j,k-1) - V_n(i,j,k-1) + V_n(i,j,(k-2)^+) + V_n(i,j,k)] \\ &\leq \mu_2[V_n(i-1,j,(k-2)^+) + V_n(i,j,k)], \end{aligned}$$

where the first inequality follows from (A.16) and the second from (5.10). We assumed that $i>0$ and $k>0$. When $j>0$, by Lemma 5.2.1 we know that the optimal action is to serve classes 1 and 2 in state (i,j,k). Situation 4 means that it is optimal to serve class 0 in state (i,j,k). Therefore we can assume that $j=0$ in this situation.

The terms in the RHS of (A.15) with a factor μ_1 are:

$$\mu_1 V_n(i,j,k-1) + \mu_1 V_n(i-1,(j-1)^+,k). \tag{A.17}$$

By (5.10) we have that (A.17) with $j=0$ is smaller than or equal to $\mu_1 V_n(i,0,k) + \mu_1 V_n(i-1,0,k-1)$. Together this gives that (A.15) is not larger than (A.14), which was to be proved.

Proof of inequality (5.11)
Inequality (5.11) can be proved in the same way as (5.10).

Proof of inequality (5.12)
We have to show that $W = \tilde{V}_{n+1}$ satisfies (5.12), that is

$$\tilde{V}_{n+1}(i,(j-1)^+,(k-1)^+) + \tilde{V}_{n+1}(i,j,k) \leq \tilde{V}_{n+1}(i,(j-1)^+,k) + \tilde{V}_{n+1}(i,j,(k-1)^+). \tag{A.18}$$

For $j=0$ or $k=0$, this is trivially true, so from now on we assume that $j>0$ and $k>0$. The value of the RHS of (A.18) depends on which actions are optimal in the two states $(i,j-1,k)$ and $(i,j,k-1)$. This means that there are four combinations to be considered in which the RHS of (A.18) has the following values:

$$\begin{array}{ll} \text{Situation 1:} & \mu_0 V_n(i-1,j-1,k) + \mu_1 V_n(i,j-1,k) + \mu_2 V_n(i,j-1,k) \\ & +\mu_0 V_n((i-1)^+,j,k-1) + \mu_1 V_n(i,j,k-1) + \mu_2 V_n(i,j,k-1), \\ \text{Situation 2:} & \mu_0 V_n(i,j-1,k) + \mu_1 V_n(i,(j-2)^+,k) + \mu_2 V_n(i,j-1,k-1) \\ & +\mu_0 V_n(i,j,k-1) + \mu_1 V_n(i-1,j-1,k-1) + \mu_2 V_n(i,j,(k-2)^+), \\ \text{Situation 3:} & \mu_0 V_n(i,j-1,k) + \mu_1 V_n(i,(j-2)^+,k) + \mu_2 V_n(i,j-1,k-1) \\ & +\mu_0 V_n((i-1)^+,j,k-1) + \mu_1 V_n(i,j,k-1) + \mu_2 V_n(i,j,k-1), \\ \text{Situation 4:} & \mu_0 V_n(i-1,j-1,k) + \mu_1 V_n(i,j-1,k) + \mu_2 V_n(i,j-1,k) \\ & +\mu_0 V_n(i,j,k-1) + \mu_1 V_n(i,j-1,k-1) + \mu_2 V_n(i,j,(k-2)^+). \end{array} \tag{A.19}$$

For each situation we will show that (A.18) holds.

- Situation 1&2: The inequality follows from (5.12).

- Situation 3: When $i = 0$, we can never be in this situation, because then serving class 0 in state $(i, j, k-1)$ is not the minimizer, since V_n is increasing. Therefore we assume $i > 0$. We can write

$$\begin{aligned}
&\tilde{V}_{n+1}(i, j-1, k-1) + \tilde{V}_{n+1}(i, j, k) \\
&\leq \mu_0 V_n(i-1, j-1, k-1) + \mu_1 V_n(i, j-1, k-1) + \mu_2 V_n(i, j-1, k-1) \\
&\quad + \mu_0 V_n(i, j, k) + \mu_1 V_n(i, j-1, k) + \mu_2 V_n(i, j, k-1). \qquad \text{(A.20)}
\end{aligned}$$

By (5.6), the terms in (A.20) with factor μ_0 or μ_1 are not larger than

$$\begin{aligned}
&-\mu_0 V_n(i-1, j-1, k) + \mu_0 V_n(i-1, j-1, k-1) + \mu_1 V_n(i, j-1, k-1) - \mu_1 V_n(i, j-1, k) \\
&+\mu_0 V_n(i, j-1, k) + \mu_0 V_n(i-1, j, k) + \mu_1 V_n(i, (j-2)^+, k) + \mu_1 V_n(i, j, k) \\
&= \mu_0 V_n(i, j-1, k) + \mu_1 V_n(i, (j-2)^+, k) + \mu_0 V_n(i-1, j, k-1) + \mu_1 V_n(i, j, k-1) \\
&\quad + \mu_0 V_n(i-1, j, k) + \mu_0 V_n(i-1, j-1, k-1) - \mu_0 V_n(i-1, j-1, k) - \mu_0 V_n(i-1, j, k-1) \\
&\quad + \mu_1 V_n(i, j, k) + \mu_1 V_n(i, j-1, k-1) - \mu_1 V_n(i, j-1, k) - \mu_1 V_n(i, j, k-1) \\
&\leq \mu_0 V_n(i, j-1, k) + \mu_1 V_n(i, (j-2)^+, k) + \mu_0 V_n(i-1, j, k-1) + \mu_1 V_n(i, j, k-1),
\end{aligned}$$

where the inequality follows from (5.12), since $\mu_0 V_n(i-1, j, k) + \mu_0 V_n(i-1, j-1, k-1) \leq \mu_0 V_n(i-1, j-1, k) + \mu_0 V_n(i-1, j, k-1)$ and $\mu_1 V_n(i, j, k) + \mu_1 V_n(i, j-1, k-1) \leq \mu_1 V_n(i, j-1, k) + \mu_1 V_n(i, j, k-1)$. Together this gives that (A.20) is not larger than (A.19), which was to be proved.

- Situation 4: This is symmetric to situation 3. □