Ann Oper Res (2009) 170: 59–78
DOI 10.1007/s10479-008-0442-y

Interacting queues with server selection and coordinated scheduling—application to cellular data networks

Sem Borst · Nidhi Hegde · Alexandre Proutière

Published online: 24 September 2008

Abstract We consider a system of parallel servers handling users of various classes, whose service rates depend not only on user classes, but also on the set of active servers. We investigate the stability under two types of allocation strategies: (i) server assignment where the users are assigned to servers based on rates, load, and other considerations, and (ii) coordinated scheduling where the activity states of servers are coordinated. We show how the model may be applied to evaluate the downlink capacity of wireless data networks. Specifically, we examine the potential gains in wireless capacity from the two types of resource allocation strategies.

Keywords Load balancing · Coordinated scheduling · Cellular networks

We investigate the stability of a fairly general system of parallel servers handling users of various classes. The service rate of a user depends not only on the class of the user and the server involved, but also on the set of active servers. Users of the various classes enter the system according to some stationary ergodic processes and leave the system after having been served. We focus on two types of coordinated resource allocation strategies: (i) *server assignment* and (ii) *coordinated scheduling*.

The motivation for *server assignment* arises from the natural principle that the overall performance may be improved by optimizing the allocation of users to servers. This may be

S. Borst (✉)
Alcatel-Lucent Bell Labs, P.O. Box 636, Murray Hill, NJ 07974-0636, USA
e-mail: sem@alcatel-lucent.com

S. Borst
Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands

N. Hegde
Orange Labs, 38-40 rue du Général Leclerc, 92794 Issy-les-Moulineaux, France

A. Proutière
Microsoft Research, 7JJ Thomson Avenue, Cambridge CB30FB, UK

achieved by assigning users to servers not solely based on service rates, but taking load and other relevant considerations into account as well. It is worth emphasizing that in the system considered, the optimal assignment of users to servers is fundamentally different from a standard load balancing problem in two crucial respects. First of all, since service rates are highly server-dependent, moving traffic from servers offering high rates to those with lower rates imposes extra load. This is similar to the concept of *server affinity* in the processor scheduling literature, where each incoming job has some default or preferred server, and a certain overhead or penalty is incurred when transferring a job to an alternative server, see for instance Squillante et al. (2001). Secondly, in the system we consider, the service rates for a given user do not only vastly differ among servers, they also strongly depend on which subset of the servers is active. This interaction between the server capacities implies that in general, it is not optimal to perfectly balance the load among servers.

The principle of *coordinated scheduling* is to decide, at each scheduling instant in a centralized manner, which servers should be active and which users they should serve. The rationale for coordinated scheduling stems from the simple fact that the service rates are impacted by the activity states of other servers, so that significantly higher rates may be achieved when certain servers are switched off. When the increase in the rates is sufficiently large, it may outweigh the sacrifice of resources at the servers that are turned off, yielding a net benefit.

Our primary interest is in studying the stability of the system when server assignment and/or coordinated scheduling is allowed. With server assignment only, we can provide capacity-optimal schemes and exactly characterize the stability region when the server capacities do not depend on the set of active servers or when the system reduces to two servers. In other cases, we provide conservative estimates of the stability region that can be achieved using server assignment. When coordinated scheduling is allowed, either in the absence of or in addition to server assignment, the stability region can be completely characterized.

The analysis of such systems is motivated by the performance evaluation of the downlink of cellular networks supporting data traffic. In such networks, file downloads are randomly generated by clients, and cease upon transfer completion. Base stations (BS's) are assigned to transfer these files, and the transmission rates from the given BS's depend not only on the position of the corresponding clients (assumed to be fixed here over the duration of a transfer), but also on the interference generated by the other BS's (interference is only generated by a BS if it is active). In our model, BS's correspond to servers, and the clients' possible positions in the network to user classes. In data networks user-perceived performance is mainly determined by the file transfer time. A primary criterion to ensure that the transfer time is finite is the stability of the system. As in Bonald and Proutière (2003) we use the notion of *network capacity*, the maximum amount of traffic that can be supported for a given spatial traffic pattern. For a few illustrative examples of wireless networks, we will examine the potential capacity gains from coordinated resource allocation strategies.

The concepts of coordinated scheduling and server assignment have each been considered in isolation before. The notion of coordinated scheduling has been exploited in the context of wireless data networks in Bonald et al. (2005), while the role of server assignment in load balancing and capacity maximization has been investigated in Das et al. (2003), Bonald et al. (2004c), Sang et al. (2004). In the present paper we will focus on the combination of these two concepts. In particular, we will examine the relative merits of both features as a function of the network topology.

The remainder of the paper is organized as follows. In Sect. 1 we introduce the model and describe the resource allocation strategies considered. In Sect. 2 we determine the system capacity with server assignment only. We examine the system capacity with coordinated

scheduling in Sect. 3. In Sect. 4 we apply our model to wireless data networks and discuss the numerical experiments that we conducted to support the analytical findings. We provide concluding remarks in Sect. 5.

1 Model description and resource allocation strategies

1.1 The model

We consider a set of servers indexed by $\mathcal{N} = \{1, \ldots, N\}$ which are shared by a dynamic population of users of various classes labeled by the set $\mathcal{X} = \{1, \ldots, X\}$. An important feature of the model is that the queues are interacting in the sense that the service rate of a particular server depends on the activities of other servers. Specifically, we denote by $C_{nx,\mathcal{A}}$ the service rate of a class-x user when served by server n and when the set of active servers is $\mathcal{A} \subseteq \mathcal{N}$. We assume that the service rates $C_{nx,\mathcal{A}}$ satisfy the following natural monotonicity property:

$$\forall \mathcal{A} \subseteq \mathcal{B},\ \forall n, x, \quad C_{nx,\mathcal{A}} \geq C_{nx,\mathcal{B}}. \tag{1}$$

Class-x users arrive according to a Poisson process of intensity λ_x and have i.i.d. exponentially distributed service requirements with mean σ_x. We denote by $\rho_x = \lambda_x \times \sigma_x$ the traffic load associated with class-x users, and by $p_x = \rho_x/\rho$ the proportion of the total traffic load $\rho = \sum_{x \in \mathcal{X}} \rho_x$. The model and the presented results can be easily generalized to the case of renewal processes for arrivals and service requirements.

1.2 Resource allocation

The resource allocation problem consists in determining when a given user should be served by a certain server. In the present paper, we consider three scenarios that differ in terms of the allowable resource allocation strategies.

Case 1. Server assignment only (SA) In the first scenario, there is no scheduling coordination between the servers: if a given server has a user to serve in its queue, it will be active. Each server is assumed to serve users in its queue according to some work-conserving service discipline. When a class-x user arrives, it is assigned to one of the queues according to some server selection strategy, possibly depending on the state of the system, for the entire duration of service. Denote by SA the set of all server assignment strategies.

Case 2. Coordinated scheduling only (CS) In the second scenario, users of a given class are always assigned to the same server. Denote by $\mathcal{X}_n$ the set of user classes assigned to server n. $(\mathcal{X}_n)_{n \in \mathcal{N}}$ constitutes a partition of $\mathcal{X}$. The service capabilities can be described by a set of service profiles $\mathcal{J} = \{1, \ldots, J\}$. Each of the profiles corresponds to a particular allocation of the server resources among the various user classes: a transmission profile j is determined by a set $\mathcal{A}$ of active servers and a set of user classes $\{x_n \in \mathcal{X}_n, n \in \mathcal{A}\}$ served by these active servers. We denote by $R_{x,j}$ the service rate of class-x users when profile j is used. Thus $R_{x,j} = C_{nx,\mathcal{A}}$ if $n \in \mathcal{A}$ and $x_n = x$, $R_{x,j} = 0$ otherwise. At any time, one service profile can be selected for operating the system. The transmission profiles are selected according to some scheduling policy and may or may not depend on the system state. We denote by CS the set of coordinated scheduling strategies.

Case 3. Combined server assignment and scheduling (SACS) In the third scenario, we combine the coordinated scheduling and server assignment capabilities. Again, the service capabilities can be described by a set of service profiles $\mathcal{H} = \{1, \ldots, H\}$, with each of the profiles corresponding to a particular allocation of the server resources among the various user classes: a service profile is determined by a set $\mathcal{A}$ of active servers and a set of user classes $\{x_n \in \mathcal{X}, n \in \mathcal{A}\}$ served by the active servers (note that the only difference with the second scenario is that server n may serve any user class, $x_n \in \mathcal{X}$ instead of $x_n \in \mathcal{X}_n$). At any time, one transmission profile can be selected for operating the network. As in the second scenario, the service profiles are selected according to some scheduling policy, and may or may not depend on the system state. We denote by SACS the collection of resource allocation strategies applying both coordinated scheduling and server selection.

1.3 System stability and capacity

We describe the system state by a stochastic process $\{Z(t)\}_{t\geq 0}$. Depending on the resource allocation scenario considered, $Z(t)$ includes enough information for the corresponding process to be Markovian. In scenario 1 (SA only), $Z(t) = (Q_{n,x}(t), n \in \mathcal{N}, x \in \mathcal{X})$, where $Q_{n,x}(t)$ is the number of class-x users in the queue associated with server n at time t. In scenarios 2 and 3 (with coordinated scheduling), $Z(t) = (Q_x(t), x \in \mathcal{X})$, where $Q_x(t)$ is the total number of class-x users in the system at time t.

For a given scenario, we are interested in the stability of the system, i.e., the positive recurrence of the Markov process $\{Z(t)\}_{t\geq 0}$, depending on the resource allocation strategy π considered. For any fixed load distribution among classes $(p_x)_{x\in\mathcal{X}}$, we define the capacity of the strategy π as the maximum total traffic load C_π such that for all $\epsilon > 0$, the system is stable under π when the traffic load is $\mathsf{C}_\pi - \epsilon$. For any set of strategies $\mathcal{S}$, let $\mathsf{C}_\mathcal{S} = \max_{\pi\in\mathcal{S}} \mathsf{C}_\pi$. We further define the stability region under the set of strategies $\mathcal{S}$ as the X-dimensional set of traffic loads $(\rho_x)_{x\in\mathcal{X}}$ such that there exists a strategy $\pi \in \mathcal{S}$ stabilizing the system.

2 Capacity with server assignment only

In this section we analyze the system capacity when smart server assignment strategies can be used only. A resource allocation scheme is then determined by the work-conserving discipline used by each server and by the server assignment strategy.

2.1 No server interaction

We first discuss the case of non-interacting servers, where the service rates of users at a given server do not depend on the activity states of the other servers. In other words, for all $x \in \mathcal{X}, n \in \mathcal{N}, \mathcal{A} \subseteq \mathcal{N}, C_{nx,\mathcal{A}} = C_{nx}$.

Define $\mathcal{T}^N = \{\alpha \in \mathbb{R}_+^N : \sum_{n=1}^N \alpha_n \geq 1\}$ and $\mathcal{T}^X = \{\beta \in \mathbb{R}_+^X : \sum_{x=1}^X \beta_x \leq 1\}$. Also, define

$$\mathcal{R}_{\mathrm{SA}} = \left\{ r \in \mathbb{R}_+^{\mathcal{X}} : \forall x, \exists \alpha_x \in \mathcal{T}^N \text{ s.t. } \forall n, \sum_{x\in\mathcal{X}} \alpha_{x,n} r_x / C_{nx} \leq 1 \right\}.$$

The variable $\alpha_{x,n}$ may be interpreted as the fraction of class-x users assigned to server n. Taking $\beta_{n,x} = \alpha_{x,n} r_x / C_{nx}$, it is easily seen that $\mathcal{R}_{\mathrm{SA}}$ may be equivalently defined as

$$\mathcal{R}_{\mathrm{SA}} = \left\{ r \in \mathbb{R}_+^{\mathcal{X}} : \forall x, \exists \beta_n \in \mathcal{T}^X \text{ s.t. } \forall x, r_x \leq \sum_{n=1}^N \beta_{n,x} C_{nx} \right\}.$$

The variable $\beta_{n,x}$ may be interpreted as the fraction of resources allocated to class-x users at server n. The next proposition provides the capacity of server assignment strategies for any work-conserving discipline.

Proposition 1 *For any work-conserving discipline used at each server, we have*:

$$\mathsf{C}_{\mathrm{SA}} = \max\{\rho : \rho \times (p_1, \ldots, p_X) \in \mathcal{R}_{\mathrm{SA}}\}. \tag{2}$$

Furthermore there exists a set of vectors $\alpha_x \in \mathcal{T}^N$, $x \in \mathcal{X}$ such that:

$$\mathsf{C}_{\mathrm{SA}} = \left(\sum_{x\in\mathcal{X}} \frac{\alpha_{x,n} p_x}{C_{nx}}\right)^{-1}, \quad \forall n \in \mathcal{N}. \tag{3}$$

Finally, fix the load distribution $(p_x)_{x\in\mathcal{X}}$. Any static server assignment strategy defined by vectors $\alpha_x \in \mathcal{T}^N$, $x \in \mathcal{X}$, that stabilizes the system for all $\rho < \mathsf{C}_{\mathrm{SA}}$, satisfies (3).

Observe that (3) simply reflects the fact that under the capacity-maximizing assignment the loads of all servers are equal to one.

Proof The proof follows from the interpretation of the coefficients α and β involved in the definition of $\mathcal{R}_{\mathrm{SA}}$. Indeed, assume that the server assignment strategy achieves stability. Then define $\alpha_{x,n}$ as the stationary proportion of class-x users assigned to server n. We must have $\alpha_x \in \mathcal{T}^N$. Furthermore, the load of each server has to be less than 1, i.e., $\sum_{x\in\mathcal{X}} \alpha_{x,n}\rho p_x / C_{nx} \le 1$. This implies that $\rho = \max\{u : u \times (p_1, \ldots, p_X) \in \mathcal{R}_{\mathrm{SA}}\}$. Conversely, if the previous inequality holds, we can simply construct a static server assignment strategy stabilizing the system.

Now the proof of (3) involves a swapping argument. Define the vectors $\alpha_x \in \mathcal{T}^N$ such that for all n, $\mathsf{C}_{\mathrm{SA}} \times \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n} p_x}{C_{nx}} \le 1$. Also, define $n_1 = \arg\max_{n\in\mathcal{N}} \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n} p_x}{C_{nx}}$ and $n_2 = \arg\max_{n\in\mathcal{N}\setminus\{n_1\}} \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n} p_x}{C_{nx}}$. Now assume that for $\delta > 0$, $\sum_{x\in\mathcal{X}} \frac{\alpha_{x,n_1} p_x}{C_{n_1 x}} > \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n_2} p_x}{C_{n_2 x}} + \delta$. Now define for a given x, $\alpha'_{x,n} = \alpha_{x,n}$, for all $n \neq n_1, n_2$, and $\alpha'_{x,n_1} = \alpha_{x,n_1} - \epsilon$, $\alpha'_{x,n_2} = \alpha_{x,n_1} + \epsilon$, where $\epsilon = \delta \min(C_{n_1 x}, C_{n_2 x})/(2p_x)$. Then $\alpha'_x \in \mathcal{T}^N$, and by definition of n_1, n_2, for all n, $\mathsf{C}_{\mathrm{SA}}(1+\nu) \times \sum_{x\in\mathcal{X}} \frac{\alpha'_{x,n} p_x}{C_{nx}} \le 1$, where

$$\nu = \epsilon \min_{n\in\mathcal{N}} \left(\frac{p_x \min(1/C_{n_1 x}, 1/C_{n_2 x})}{\sum_{x\in\mathcal{X}} \frac{\alpha_{x,n} p_x}{C_{nx}}} \right) > 0.$$

This contradicts the definition of C_{SA}. Applying this argument recursively, we obtain (3). □

The above results imply that the stability region of server assignment strategies is $\check{\mathcal{R}}_{\mathrm{SA}}$, the largest open subset of $\mathcal{R}_{\mathrm{SA}}$, and that static server assignment strategies suffice to achieve the full stability region and thus the network capacity. From a practical perspective, however, static strategies are less useful, since there generally exists no single server assignment strategy that guarantees stability whenever possible. Determining the specific server assignment strategy that achieves stability for a given $(\rho_x)_{x\in\mathcal{X}} \in \check{\mathcal{R}}_{\mathrm{SA}}$ requires detailed information on the traffic loads of the various user classes, which is typically impractical to obtain.

In contrast, there do exist simple and parsimonious dynamic load balancing schemes that are more robust and ensure stability for all $(\rho_x)_{x\in\mathcal{X}} \in \check{\mathcal{R}}_{\mathrm{SA}}$ without any explicit knowledge of the actual traffic loads. Denote by $V_n(t) = \sum_{x\in\mathcal{X}} Q_{n,x}(t)\sigma_x / C_{nx}$ the *'workload'* of server n

at time t. In Stolyar (2005), the MinDrift server assignment scheme is proposed and proved to achieve stability for any $\rho \in \check{\mathcal{R}}_{\mathrm{SA}}$, irrespective of the chosen work-conserving discipline considered. The MinDrift rule assigns a class-x user arriving at time t to server $n_x^*(t)$ where:

$$n_x^*(t) = \arg \min_{n=1,\ldots,N} V_n(t)/C_{nx}.$$

Note that unfortunately the above scheme depends on the mean service requirements σ_x of the various user classes, which is usually information that is not available. The first natural dynamic server assignment strategy that does not require the information on the mean service requirements is the Join-the-Shortest-Queue policy (JSQ). This policy achieves maximum stability when the service rate at a given server does not depend on the user class (Foley and McDonald 2001). It was proved in Chernova and Foss (1998) that unfortunately it does not stabilize the system in general for class-dependent service rates, when the discipline at each server is FCFS. The stability of the JSQ policy in conjunction with other disciplines remains largely unknown.

2.2 With server interaction

We now examine the case where the service rates at a given server depend on the activity states of the other servers. As observed in Bonald et al. (2004a), the intricate correlation among the various servers renders an exact analysis elusive in general. In the special case of a static server assignment strategy and each server handling a single user class, the system corresponds to a so-called coupled-processors model, which even in the case of two queues is barely tractable, although the stability condition is then relatively simple.

Fayolle and Iasnogorodski (1979) showed that in the case of exponentially distributed service requirements the analysis of the joint queue length distribution may be formulated as a Riemann-Hilbert problem. Cohen and Boxma (1983) considered the case of generally distributed service requirements, and showed that the joint workload distribution may be obtained as the solution to a boundary value problem.

The fact that even the single-class two-queue case is nearly intractable, testifies to the complexity of the model in general. Even for three queues, hardly any results are known (Cohen 1984). In order to illustrate the complications, let us inspect the case of three servers and three user classes with $\mathcal{X}_n = \{n\}$, $n = 1, 2, 3$. We assume that the service rates of servers 1 and 2 are not affected by the activity states of the other servers, while the service rates of server 3 do depend on the activity states of both servers 1 and 2. If $\rho_1 < C_{11}$, $\rho_2 < C_{22}$, then servers 1 and 2 are guaranteed to be stable, and the necessary and sufficient condition for stability of server 3 may be expressed as

$$\rho_3 < \pi_{00} C_{33,\{3\}} + \pi_{01} C_{33,\{1,3\}} + \pi_{10} C_{33,\{2,3\}} + \pi_{11} C_{33,\{1,2,3\}},$$

with π_{ij} representing the probability that servers i and j are in states i and j, where 0 and 1 stand for inactive and active, respectively. Thus, in order to provide stability guarantees, a server assignment strategy must be able to maximize the latter expression, which is a quite a challenging task. Although $\pi_{10} + \pi_{11} = \rho_1/C_{11}$ and $\pi_{01} + \pi_{11} = \rho_2/C_{22}$, the individual probabilities π_{ij} and thus the value of the entire expression depend on the detailed features of the server assignment scheme. (If the service rates of servers 1 and 2 did depend on the activity states of the other servers, then the probabilities in fact may even be sensitive to the service requirement distributions (Bonald et al. 2004a).) This suggests that static assignment schemes will in general not suffice to achieve the full stability region and system capacity, and that there do not even exist any simple greedy dynamic schemes that do so.

In view of the above observations, we focus in the remainder of the section on the two-server case where we perform an exact stability analysis and compute the capacity C_{SA} of the system under server assignment strategies, and on the derivation of conservative bounds for the capacity of systems with more than two servers.

2.2.1 Two servers

For compactness, denote $C_{nx,\mathrm{on}} = C_{nx,\{1,2\}}$ and $C_{nx,\mathrm{off}} = C_{nx,\{n\}}$, $n = 1, 2$.

The impact of the service discipline on the stability region In the case of a system with two servers only and a static server assignment strategy, the stability depends in general on the work-conserving discipline used at the two servers. Define two permutations σ_1 and σ_2 of $\mathcal{X}$ such that for $n = 1, 2$,

$$\frac{C_{n\sigma_n(1),\mathrm{off}}}{C_{n\sigma_n(1),\mathrm{on}}} \leq \cdots \leq \frac{C_{n\sigma_n(X),\mathrm{off}}}{C_{n\sigma_n(X),\mathrm{on}}}.$$

Consider a static server assignment strategy that assigns a fraction $\alpha_{x,n}$ of the class-x users to server n with $\sum_{n=1}^{2} \alpha_{x,n} = 1$ for all $x \in \mathcal{X}$. Define the load of server n when the other server is active by $\rho_{n,\mathrm{on}}(\alpha) = \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n}\rho_x}{C_{nx,\mathrm{on}}}$. Also, define the following partial loads: $\rho^x_{n,\mathrm{on}}(\alpha) = \sum_{k=1}^{x} \frac{\alpha_{\sigma_n(k),n}\rho_{\sigma_n(k)}}{C_{n\sigma_n(k),\mathrm{on}}}$ and $\rho^x_{n,\mathrm{off}}(\alpha) = \sum_{k=x}^{X} \frac{\alpha_{\sigma_n(k),n}\rho_{\sigma_n(k)}}{C_{n\sigma_n(k),\mathrm{off}}}$. The following proposition identifies the work-conserving disciplines at each server maximizing the stability region.

Proposition 2 *Consider the service discipline at server n such that*:

(i) *when server $m \neq n$ is active, users are served with priority decreasing with $\sigma_n(x)$,*
(ii) *when server $m \neq n$ is inactive, users are served with priority increasing with $\sigma_n(x)$.*

The above service discipline maximizes the stability region. The load vector (ρ_1, ρ_2) belongs to the latter region if for $n = 1$ or for $n = 2$:

$$\rho_{n,\mathrm{on}}(\alpha) < 1,$$

and there exists $x*$ such that: for $m \neq n$,

$$\rho^{x*+1}_{m,\mathrm{off}}(\alpha) \leq 1 - \rho_{n,\mathrm{on}}(\alpha) < \rho^{x*}_{m,\mathrm{off}}(\alpha),$$

$$\rho^{x*-1}_{m,\mathrm{on}}(\alpha) + \frac{\alpha_{\sigma_m(x*),m}\rho_{\sigma_m(x*)}}{C_{m\sigma_m(x*),\mathrm{on}}} - (1 - \rho_{n,\mathrm{on}}(\alpha) - \rho^{x*+1}_{m,\mathrm{off}}(\alpha))\frac{C_{m\sigma_m(x*),\mathrm{off}}}{C_{m\sigma_m(x*),\mathrm{on}}} < \rho_{n,\mathrm{on}}(\alpha).$$

The proof of the above result is similar to that of the next proposition studying the stability in case of the Processor-Sharing discipline. The fact that the service discipline considered is optimal is just based on the principle that one should be as opportunistic as possible and exploit the inactive periods of the other server by serving users that benefit the most from this inactivity, Specifically, classes $1, \ldots, x^* - 1$ suffer relatively little from activity (benefit relatively little from inactivity) of the other server, and are hence served when the other server is active, while classes $x^* + 1, \ldots, X$ are affected more (benefit less), and are served when the other server is inactive.

Processor-Sharing discipline We now investigate the system stability under the Processor-Sharing (PS) discipline. Consider a static strategy that assigns a fraction $\alpha_{x,n}$ of the class-x users to server n with $\sum_{n=1}^{N} \alpha_{x,n} = 1$ for all $x \in \mathcal{X}$. Assuming $\rho_{n,\{1,2\}}(\alpha) \leq 1$, define $\overline{C}_{n'x}(\alpha) = \rho_{n,\mathrm{on}}(\alpha) C_{n'x,\mathrm{on}} + (1 - \rho_{n,\mathrm{on}}(\alpha)) C_{n'x,\mathrm{off}}$, $n' \neq n$. Note that $\overline{C}_{n'x}(\alpha)$ may be interpreted as the average transmission rate of class-x users at server n' given that server n is active a fraction of the time $\rho_{n,\mathrm{on}}(\alpha) \leq 1$. Define $\mathcal{R}(\alpha)$ as the set of all vectors $(\rho_1, \ldots, \rho_X) \in \mathbb{R}_+^{\mathcal{X}}$ such that

$$\rho_{1,\mathrm{on}}(\alpha) \leq 1 \quad \text{and} \quad \sum_{x \in \mathcal{X}} \frac{\alpha_{x,2}\rho_x}{\overline{C}_{2x}(\alpha)} \leq 1, \quad \text{or} \quad \rho_{2,\mathrm{on}}(\alpha) \leq 1 \quad \text{and} \quad \sum_{x \in \mathcal{X}} \frac{\alpha_{x,1}\rho_x}{\overline{C}_{1x}(\alpha)} \leq 1.$$

The next proposition provides a necessary and sufficient stability condition.

Proposition 3 *A static server assignment strategy achieves stability if* $(\rho_x)_{x \in \mathcal{X}} \in \check{\mathcal{R}}(\alpha)$. *If* $(\rho_x)_{x \in \mathcal{X}} \notin \mathcal{R}(\alpha)$, *the system is unstable.*

Proof First assume that $\rho_{1,\{1,2\}}(\alpha) < 1$ and $\sum_{x \in \mathcal{X}} \frac{\alpha_{x,2}\rho_x}{\overline{C}_{2x}(\alpha)} < 1$. Due to the monotonicity property (1), we have $Q_{n,x}(t) \leq \tilde{Q}_{n,x}(t)$, where $\tilde{Q}_{n,x}(t)$ is the number of class-x users in the queue of server n at time t in the following fictitious system. In the latter system, the arrivals are identical to those in the original system, and the service rates of class-x users at server 1 is $C_{1x,\mathrm{on}}$. The initial states in both systems are identical $\tilde{Q}_{n,x}(0) = Q_{n,x}(0)$ for all n and x. Now with the assumption on $(\rho_x)_{x \in \mathcal{X}}$, the Markov process $(\tilde{Q}_{n,x}(t), n, x)_{t \geq 0}$ is positive recurrent: $(\tilde{Q}_{1,x}(t), x)_{t \geq 0}$ corresponds to the numbers of users in a multi-class queue of load $\rho_{1,\{1,2\}}(\alpha)$ strictly less than 1 and then $(\tilde{Q}_{2,x}(t), x)_{t \geq 0}$ may be interpreted as the numbers of users in a multi-class PS queue with a capacity varying according to a stationary ergodic process independent of the state the queue. Its stability is ensured by $\sum_{x \in \mathcal{X}} \frac{\alpha_{x,2}\rho_x}{\overline{C}_{2x}(\alpha)} < 1$, see for example (Bonald et al. 2004b).

Assume now that $(\rho_x)_{x \in \mathcal{X}} \notin \mathcal{R}(\alpha)$. If $\rho_{n,\{1,2\}}(\alpha) > 1$ for $n = 1, 2$, then if at time 0, there are some users assigned to servers 1 and 2, the system evolves as two independent multi-class queues with load greater than 1 until one of the two queues empties. But there is a positive probability that neither of these queues empties, and that they both increase indefinitely, which implies instability. It remains to investigate the case where $\rho_{1,\mathrm{on}}(\alpha) \leq 1$ and $\sum_{x \in \mathcal{X}} \frac{\alpha_{x,2}\rho_x}{\overline{C}_{2x}(\alpha)} > 1$. Denote $T_0 = \inf\{t : Q_2(t) = 0\}$. We prove that T_0 is infinite with positive probability, which implies instability. For all $t < T_0$, the numbers of users $Q_{2,x}(t)$ evolve as the number of users in a multi-class PS queue with time-varying capacity defined by an independent process $Q_1(t)$ (note that we have independence only because we consider $t < T_0$). This queue has a load strictly greater than 1, which implies that T_0 is infinite with positive probability. □

It follows that the network capacity under the static assignment strategy α may be characterized as $\mathsf{C}(\alpha) = \max\{\rho : \rho(p_1, \ldots, p_X) \in \mathcal{R}(\alpha)\}$.

An interesting special case is where all the user classes assigned to server n enjoy the same relative increase K_n in transmission rate when server $n' \neq n$ is inactive, i.e., $\frac{C_{nx,\mathrm{off}}}{C_{nx,\mathrm{on}}} = K_n$ for all user classes x with $\alpha_{x,n} > 0$. In that case, $\overline{C}_{n'x}(\alpha) = C_{n'x,\mathrm{on}}(\rho_{n,\{1,2\}}(\alpha) + (1 - \rho_{n,\{1,2\}}(\alpha))K_{n'}$ for all user classes x with $\alpha_{x,n} > 0$. Thus, $\mathcal{R}(\alpha)$ may be defined as the set of all vectors such that

$$\rho_{1,\{1,2\}}(\alpha) < 1 \quad \text{and} \quad \rho_{2,\{1,2\}}(\alpha) < \rho_{1,\{1,2\}}(\alpha) + (1 - \rho_{1,\{1,2\}}(\alpha))K_2,$$

or

$$\rho_{2,\{1,2\}}(\alpha) < 1 \quad \text{and} \quad \rho_{1,\{1,2\}}(\alpha) < \rho_{2,\{1,2\}}(\alpha) + (1 - \rho_{2,\{1,2\}}(\alpha))K_1.$$

In that case, the system behaves as a coupled-processors model, and the above proposition follows from Cohen and Boxma (1983). Define S as the set of all vectors $(r_1, r_2) \in \mathbb{R}_+^2$ such that

$$r_1 \le 1 \quad \text{and} \quad r_2 \le r_1 + (1 - r_1)K_2, \quad \text{or} \quad r_2 \le 1 \quad \text{and} \quad r_1 \le r_2 + (1 - r_2)K_1.$$

Proposition 4 *In the special case just described, no server assignment scheme achieves stability unless there exists a static assignment strategy* α *such that* $(\rho_{1,\{1,2\}}(\alpha), \rho_{2,\{1,2\}}(\alpha)) \in S$.

Proof Suppose that there exists no static assignment strategy such that $(\rho_{1,\{1,2\}}(\alpha), \rho_{2,\{1,2\}}(\alpha)) \in S$. We will show that no server assignment scheme achieves stability.

We distinguish between two cases: $(K_1 - 1)(K_2 - 1) \le 1$ and $(K_1 - 1)(K_2 - 1) \ge 1$. In case $(K_1 - 1)(K_2 - 1) \le 1$, we have $S = \text{conv}(\{(0, 0), (1, 1), (K_1, 0), (0, K_2)$, and the assertion easily follows. Thus it remains to prove the statement in case $(K_1 - 1)(K_2 - 1) > 1$. In this case, we have $S = \{(r_1, r_2) \in \mathbb{R}_+^2 : r_2 \le r_1 + (1 - r_1)K_2 \text{ or } r_1 \le r_2 + (1 - r_2)K_1\}$.

We first introduce some notation. Let $Q_{n,x}(t)$ be the number of class-x flows at BS n at time t, and let $\bar{Q}_{n,x}(t)$ be the corresponding fluid limit (Dai 1995) obtained by scaling both time and the initial number of users. Denote by $V_n(t) = \sum_{x \in \mathcal{X}} \sigma_{n,x} \bar{Q}_{n,x}(t)$ the workload at server n at time t, with $\sigma_{n,x} = \sigma_x / C_{nx,\text{on}}$ the mean service requirement of a class-x user at server n. Let $A_n(s, t)$ be the amount of traffic assigned to server n during the time interval $[s, t]$ in the fluid limit, and let $B_n(s, t)$ be the amount of traffic served by server n during the time interval $[s, t]$. Then we have $V_n(t) \ge V_n(s) + A_n(s, t) - B_n(s, t)$.

We will show that $W(t) = \min\{\frac{V_1(t)}{K_1} + (1 - \frac{1}{K_1})V_2(t), (1 - \frac{1}{K_2})V_1(t) + \frac{V_2(t)}{K_2}\}$ has positive drift whenever $W(t) > 0$. The fact that there exists no static assignment strategy such that $(\rho_{1,\{1,2\}}(\alpha), \rho_{2,\{1,2\}}(\alpha)) \in S$ implies that there exists an $\epsilon > 0$ such that $\gamma_2 \ge \gamma_1 + (1 - \gamma_1)K_2 + \epsilon$ and $\gamma_1 \ge \gamma_2 + (1 - \gamma_2)K_1 + \epsilon$ for all assignment schemes with $\gamma_n = \rho_{n,\{1,2\}}(\alpha)$. This may be rewritten as $\gamma_1 + (K_1 - 1)\gamma_2 \ge K_1 + \epsilon$ and $(K_2 - 1)\gamma_1 + \gamma_2 \ge K_2 + \epsilon$. This also means that $A_1(s, t) + (K_1 - 1)A_2(s, t) \ge (K_1 + \epsilon)(t - s)$ and $(K_2 - 1)A_1(s, t) + A_2(s, t) \ge (K_2 + \epsilon)(t - s)$ for any time interval $[s, t]$ for all assignment schemes. Also, $B_1(s, t) \le K_1(t - s)$ and $B_2(s, t) \le K_2(t - s)$ for any time interval $[s, t]$.

Now suppose that $W(t) = \frac{V_1(t)}{K_1} + (1 - \frac{1}{K_1})V_2(t) = (1 - \frac{1}{K_2})V_1(t) + \frac{V_2(t)}{K_2} - w$, with $w \ge 0$. Then $V_1(t) > 0$, since $(K_1 - 1)(K_2 - 1) > 1$ implies $1 - \frac{1}{K_1} > \frac{1}{K_2}$. Denoting $V_1(t) = v_1 > 0$, we know that $V_1(t + u) > 0$ for all $u \in [0, v/K_1)$, as $V_1(t)$ cannot decrease at a rate higher than K_1, or formally

$$V_1(t + u) \ge V_1(t) + A_1(t, u) - B_1(t, u) \ge V_1(t) - K_1 u.$$

Thus, we have $B_1(t, t + u) + (K_1 - 1)B_2(t, t + u) \le K_1 u$ for all $u \in [0, v/K_1)$. Combining the above inequalities, it may be shown that $\frac{V_1(t+u)}{K_1} + (1 - \frac{1}{K_1})V_2(t + u) \ge W(t) + \frac{\epsilon}{K_1} u$ for all $u \in [0, v_1/K_1)$.

By symmetry, if $W(t) = (1 - \frac{1}{K_2})V_1(t) + \frac{V_2(t)}{K_2} = \frac{V_1(t)}{K_1} + (1 - \frac{1}{K_1})V_2(t) - w$, with $w \ge 0$, then $(1 - \frac{1}{K_2})V_1(t + u) + \frac{V_2(t+u)}{K_2} \ge W(t) + \frac{\epsilon}{K_2} u$ for all $u \in [0, v_2/K_2)$, with $v_2 = V_2(t) > 0$.

It can then be shown that there exists an $v > 0$ such that $W(t + v) \ge W(t) + v\epsilon \min\{\frac{1}{K_1}, \frac{1}{K_2}\}$, with v in fact being at least a fixed fraction of $W(t)$. It then follows that no server assignment scheme achieves stability. □

The above proposition implies that the capacity under server assignment strategies is given by:

$$\mathsf{C}_{\mathrm{SA}} = \max\left\{\rho : \rho \times (p_1, \ldots, p_X) \in \bigcup_{\alpha} \mathcal{R}(\alpha)\right\}. \tag{4}$$

2.2.2 Sufficient stability conditions

When there are more than two servers, the exact stability condition can not be established. Instead, we provide conservative sufficient stability conditions, leading to lower bounds for C_{SA}. We apply the following method: (i) first for a fixed assignment of users among servers (i.e., for fixed proportions $\alpha_{x,n}$ of class-x users served by server n), we derive a lower bound for the capacity, (ii) then we identify the optimal static assignment strategy leading to the maximum lower bound for the capacity.

Let us fix the server assignment: let $\alpha_{x,n}$ denote the proportion of class-x users assigned to server n. Define the load of server n when all servers are active by:

$$\rho_{n,\mathcal{N}}(\alpha) = \sum_{x\in\mathcal{X}} \frac{\rho\alpha_{x,n}p_x}{C_{nx,\mathcal{N}}}.$$

A sufficient condition for stability is that for all n, $\rho_{n,\mathcal{N}}(\alpha) < 1$. This condition leads to a lower bound for the capacity, referred to as *first-degree bound* in Bonald et al. (2004a):

$$\mathsf{C}(\alpha) \geq \left(\max_{n\in\mathcal{N}} \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n}p_x}{C_{nx,\mathcal{N}}}\right)^{-1}. \tag{5}$$

The proof is similar to that of Proposition 3, observing that $Q_{n,x}(t) \leq \tilde{Q}_{n,x}(t)$, where $\tilde{Q}_{n,x}(t)$ is the queue length obtained when the class-x users are served at the minimum rate $C_{nx,\mathcal{N}}$. The above result naturally leads to the following bound for C_{SA}.

$$\mathsf{C}_{\mathrm{SA}} \geq \max_{\alpha}\left(\max_{n\in\mathcal{N}} \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n}p_x}{C_{nx,\mathcal{N}}}\right)^{-1}. \tag{6}$$

We now derive tighter capacity bounds referred to as *second-degree bounds* in Bonald et al. (2004a). Again let us fix the server assignment α. Define $\overline{C}_{nx}(\alpha)$ by:

$$\overline{C}_{nx}(\alpha) = \sum_{\mathcal{A}\subseteq\mathcal{N}\setminus\{n\}} C_{nx,\mathcal{A}\cup\{n\}} \left(\prod_{m\in\mathcal{A}} (\rho_{m,\mathcal{N}}(\alpha) \wedge 1) \times \prod_{m\notin(\mathcal{A}\setminus\{n\})} (1 - (\rho_{m,\mathcal{N}}(\alpha) \wedge 1))\right).$$

The above definition may be interpreted as follows. The term $C_{nx,\mathcal{A}\cup\{n\}}$ represents the service rate received by class x from server n when the set of active servers is $\mathcal{A}\cup\{n\}$. Recall that $(\rho_{m,\mathcal{N}}(\alpha)$ is the load of server n when all servers are busy, and thus $\rho_{m,\mathcal{N}}(\alpha) \wedge 1$ provides an upper bound for the fraction of time that server n is busy. Hence, noting that $1 - (\rho_{m,\mathcal{N}}(\alpha) \wedge 1) = 0$ when $\rho_{m,\mathcal{N}}(\alpha) \geq 1$, we deduce that the term in brackets provides a lower bound for the fraction of time that the active set of servers is $\mathcal{A}$. It follows that the sum provides a lower bound for the time-average service rate received by class x.

Proposition 5 *If for all $n \in \mathcal{N}$, $\sum_{x\in\mathcal{X}} \frac{\rho\alpha_{x,n}p_x}{\overline{C}_{nx}(\alpha)} < 1$, then the system is stable.*

Proof The proof is obtained using sample path comparisons. Let us consider server 1 and analyze its stability. We can construct a fictitious system with the same initial state as the original system where the numbers $\tilde{Q}_n(t)$ of users assigned to servers $2,\ldots,N$ vary independently, and such that for all t, $Q_n(t) \leq \tilde{Q}_n(t)$, for all $n \geq 2$. This system is obtained assuming that class-x users are served at minimum rate $C_{nx,\mathcal{N}}$ by server $n \geq 2$. Then, in the fictitious system, class-x users are served at a smaller rate than in the original system. The ergodic mean of this rate is $\overline{C}_{1x}(\alpha)$, so if $\sum_{x\in\mathcal{X}} \frac{\rho\alpha_{x,1}p_x}{\overline{C}_{1x}(\alpha)} < 1$, then server 1 is stable in the fictitious system and thus in the original system as well. □

As a consequence, the system capacity $\mathsf{C}(\alpha)$ for the static server assignment strategy α satisfies:

$$\mathsf{C}(\alpha) \geq \left(\max_{n\in\mathcal{N}} \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n}p_x}{\overline{C}_{nx}(\alpha)}\right)^{-1}, \tag{7}$$

and a lower bound for the capacity under server assignment is given by:

$$\mathsf{C}_{\mathrm{SA}} \geq \max_{\alpha}\left(\max_{n\in\mathcal{N}} \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n}p_x}{\overline{C}_{nx}(\alpha)}\right)^{-1}. \tag{8}$$

3 Capacity with coordinated scheduling

In this section we determine the system capacity for the set of resource allocation strategies applying coordinated scheduling only and combining coordinated scheduling and server assignment. Denote by C_{CS} and $\mathsf{C}_{\mathrm{SACS}}$ the capacities in these two cases. It turns out that for these two types of resource allocation strategies, the condition for the system to be stable is known, see e.g. Tassiulas and Ephremides (1992), Bonald et al. (2005).

Denote by $\mathcal{T}_{\mathcal{J}}$ (resp. $\mathcal{T}_{\mathcal{H}}$) the set of vectors with real and positive components summing to 1 over $\mathcal{J}$ (resp. $\mathcal{H}$). Define the following rate regions:

$$\mathcal{R}_{\mathrm{CS}} = \left\{r : \exists\alpha \in \mathcal{T}_{\mathcal{J}} s.t. r_x \leq \sum_{j\in\mathcal{J}} \alpha_j R_{x,j} \forall x \in \mathcal{X}\right\}, \tag{9}$$

$$\mathcal{R}_{\mathrm{SACS}} = \left\{r : \exists\alpha \in \mathcal{T}_{\mathcal{H}} s.t. r_x \leq \sum_{j\in\mathcal{H}} \alpha_j R_{x,j} \forall x \in \mathcal{X}\right\}. \tag{10}$$

The previous sets are the sets of achievable rate vectors applying coordinated scheduling only and both coordinated scheduling and server assignment. The following proposition, proved in Tassiulas and Ephremides (1992) for example, characterizes the stability of networks whose resource allocation strategies are in the sets CS and SACS.

Proposition 6 (i) *If* $\rho \times (p_1,\ldots,p_X) \in \check{\mathcal{R}}_{\mathrm{CS}}$ (*resp.* $\rho \times (p_1,\ldots,p_X) \in \check{\mathcal{R}}_{\mathrm{SACS}}$), *then there exists a resource allocation strategy in CS* (*resp. SACS*) *stabilizing the system.* (ii) *In contrast, if* $\rho \times (p_1,\ldots,p_X) \notin \mathcal{R}_{\mathrm{CS}}$ (*resp.* $\rho \times (p_1,\ldots,p_X) \notin \mathcal{R}_{\mathrm{SACS}}$), *then there is no resource allocation in CS* (*resp. SACS*) *stabilizing the system.*

Note that the previous proposition holds not only for Poisson arrival processes but also for stationary ergodic arrival processes. As a consequence of these results, the system capacities

C_{CS} and $\mathsf{C}_{\mathrm{SACS}}$ may be characterized as:

$$\mathsf{C}_{\mathrm{CS}} = \max\{\rho : \rho(p_1, \ldots, p_X) \in \mathcal{R}_{\mathrm{CS}}\}, \tag{11}$$

$$\mathsf{C}_{\mathrm{SACS}} = \max\{\rho : \rho(p_1, \ldots, p_X) \in \mathcal{R}_{\mathrm{SACS}}\}. \tag{12}$$

There exist dynamic resource allocation schemes that achieve stability whenever possible and that only require knowledge of the queue lengths Q_x and the service rates $R_{x,j}$, $j \in \mathcal{J}$ (or $\mathcal{H}$). An example of such schemes is the κ-fair scheduler (Mo and Walrand 2000) choosing at time t the transmission profile j with probability α_j such that the average service rate vector $(\sum_j R_{x,j}\alpha_j, x \in \mathcal{X})$ is the unique solution of the following convex optimization problem:

$$\max \sum_{x=1}^{X} Q_x^{\kappa}(t) \frac{(\sum_j R_{x,j}\alpha_j)^{1-\kappa}}{1-\kappa}, \quad \text{s.t.} \sum_j \alpha_j \le 1.$$

For any $\kappa > 0$, it has been shown in Bonald et al. (2006) that the κ-fair scheduler achieves stability whenever possible.

3.1 Two servers

We now examine the capacity with coordinated scheduling for a two-server system. We present two results that allow us to identify the optimal static scheduling strategy for a fixed load distribution $(p_x)_{x\in\mathcal{X}}$, and to provide a simple algorithm for computing the capacities C_{CS} and $\mathsf{C}_{\mathrm{SACS}}$. The proofs of these results are similar to that of Proposition 4.1 in Bonald et al. (2005).

Using (9), (11), the network capacity with coordinated scheduling only is given by $\mathsf{C}_{\mathrm{CS}} = 1/\tau^*$, with τ^* denoting the solution of the linear program:

$$\begin{aligned}
\text{minimize} \quad & \tau = \tau_{\mathrm{on}} + \tau_{1,\mathrm{off}} + \tau_{2,\mathrm{off}} \\
\text{subject to} \quad & C_{nx,\mathrm{on}}\tau_{nx,\mathrm{on}} + C_{nx,\mathrm{off}}\tau_{nx,\mathrm{off}} \ge p_x \quad x \in \mathcal{X}_n,\ n = 1, 2, \\
& \sum_{x\in\mathcal{X}_1} \tau_{1x,\mathrm{on}} = \sum_{x\in\mathcal{X}_2} \tau_{2x,\mathrm{on}} = \tau_{\mathrm{on}} \\
& \sum_{x\in\mathcal{X}_1} \tau_{1x,\mathrm{off}} = \tau_{1,\mathrm{off}}, \qquad \sum_{x\in\mathcal{X}_2} \tau_{2x,\mathrm{off}} = \tau_{2,\mathrm{off}} \\
& \tau_{nx,\mathrm{on}}, \tau_{nx,\mathrm{off}} \ge 0 \quad x \in \mathcal{X}_n, n = 1, 2,
\end{aligned} \tag{13}$$

with $\tau_{nx,\mathrm{on}}$ and $\tau_{nx,\mathrm{off}}$ representing the amount of time that class x is served at server n while the other server is active and inactive, respectively. Without loss of generality, assume that the user classes are indexed such that $\mathcal{X}_1 = \{1, 2, \ldots, X'\}$ and $\mathcal{X}_2 = \{X'+1, X'+2, \ldots, X\}$, with

$$\frac{C_{11,\mathrm{off}}}{C_{11,\mathrm{on}}} \ge \frac{C_{12,\mathrm{off}}}{C_{12,\mathrm{on}}} \ge \cdots \ge \frac{C_{1X',\mathrm{off}}}{C_{1X',\mathrm{on}}}, \quad \text{and} \quad \frac{C_{2X'+1,\mathrm{off}}}{C_{2X'+1,\mathrm{on}}} \le \frac{C_{2X'+2,\mathrm{off}}}{C_{2X'+2,\mathrm{on}}} \le \cdots \le \frac{C_{2X,\mathrm{off}}}{C_{2X,\mathrm{on}}}.$$

The next proposition gives a characterization of the optimal solution of the above linear program, and thus of the structure of the capacity-maximizing scheduling strategy.

Proposition 7 *There exist a solution of* (13) *and* $x_1^* \le X' \le x_2^*$, $x_2^* \ge X'$ *such that the following six properties hold*: (i) $\tau_{1x,\mathrm{on}} = 1$ *for all* $x = 1, \ldots, x_1^* - 1$, (ii) $\tau_{1x_1^*,\mathrm{on}} + \tau_{1x_1^*,\mathrm{off}} = 1$,

(iii) $\tau_{1x,\text{off}} = 1$ *for all* $x = x_1^* + 1, \ldots, X'$, (iv) $\tau_{2x,\text{off}} = 1$ *for all* $x = X' + 1, \ldots, x_2^* - 1$, (v) $\tau_{2x_2^*,\text{on}} + \tau_{2x_2^*,\text{off}} = 1$, (vi) $\tau_{2x,\text{on}} = 1$ *for all* $x = x_2^* + 1, \ldots, X$.

We now turn the attention to the case with both server assignment and coordinated scheduling. Using (10), (12), the capacity $\mathsf{C}_{\text{SACS}} = 1/\tau^*$, with τ^* denoting the solution of the linear program:

$$\begin{aligned}
\text{minimize} \quad & \tau = \tau_{\text{on}} + \tau_{1,\text{off}} + \tau_{2,\text{off}} \\
\text{subject to} \quad & \sum_{n=1}^{2} C_{nx,\text{on}}\tau_{nx,\text{on}} + C_{nx,\text{off}}\tau_{nx,\text{off}} \geq p_x, \quad x \in \mathcal{X} \\
& \sum_{x\in\mathcal{X}} \tau_{1x,\text{on}} = \sum_{x\in\mathcal{X}} \tau_{2x,\text{on}} = \tau_{\text{on}} \\
& \sum_{x\in\mathcal{X}} \tau_{nx,\text{off}} = \tau_{n,\text{off}}, n = 1, 2 \\
& \tau_{nx,\text{on}}, \tau_{nx,\text{off}} \geq 0 \quad x \in \mathcal{X}_n, n = 1, 2,
\end{aligned} \tag{14}$$

with $\tau_{nx,\text{on}}$ and $\tau_{nx,\text{off}}$ representing the amount of time that class x is served at server n while the other server is active and inactive, respectively. Under the following assumption, we can, as in the case of coordinated scheduling only, identify the structure of the optimal scheduling strategy. Assume the user classes can be ordered so that:

$$\frac{C_{11,\text{off}}}{C_{11,\text{on}}} \geq \cdots \geq \frac{C_{1X,\text{off}}}{C_{1X,\text{on}}}, \quad \text{and} \quad \frac{C_{21,\text{off}}}{C_{21,\text{on}}} \leq \cdots \leq \frac{C_{2X,\text{off}}}{C_{2X,\text{on}}}.$$

This assumption is natural in wireless network models as illustrated in the next section. The next proposition characterizes the optimal solution of the above linear program. Note that Proposition 8 coincides with Proposition 7 when the *critical* class x^* is constrained to be X'.

Proposition 8 *There exist a solution of* (14) *and* $x_1^* \leq x^* \leq x_2^*$ *such that the following six properties hold*: (i) $\tau_{1x,\text{on}} = 1$ *for all* $x = 1, \ldots, x_1^* - 1$, (ii) $\tau_{1x_1^*,\text{on}} + \tau_{1x_1^*,\text{off}} = 1$, (iii) $\tau_{1x,\text{off}} = 1$ *for all* $x = x_1^* + 1, \ldots, x^*$, (iv) $\tau_{2x,\text{off}} = 1$ *for all* $x = x^* + 1, \ldots, x_2^* - 1$, (v) $\tau_{2x_2^*,\text{on}} + \tau_{2x_2^*,\text{off}} = 1$, (vi) $\tau_{2x,\text{on}} = 1$ *for all* $x = x_2^* + 1, \ldots, X$.

4 Application to wireless data networks

We now apply the results derived in the previous sections to evaluate the downlink capacity of wireless data networks. We focus on networks where each base station (BS) serves users downloading data files. The transmissions to the various users are assumed to be *orthogonal* in the sense that transmissions from the same BS do not interfere with each other. When BS activities are not coordinated, we further assume that each BS evenly shares its resources among users, i.e., it adheres to a PS discipline. Such networks are representative of CDMA 1xEV-DO (Bender et al. 2000) and UMTS-HSDPA systems for instance. In reference to the terminology used in the previous sections, a BS corresponds to a server and a data flow to a user. Flows are randomly generated by users in the network and leave once the corresponding transfer has been completed. For the sake of simplicity, we assume that users do not move during flow transfers. Flows are then classified according to the position of the corresponding users.

4.1 Radio environment

To characterize the radio environment, it is useful to introduce the notion of a feasible rate of various users. The feasible rate of a user in location x when served by BS n is defined as the rate this user would receive if all resources of BS n were allocated to this user. The feasible rate of a user depends on its Signal-to-Interference-plus-Noise ratio (SINR), which in turn depends on (i) its path loss value to the serving BS n and (ii) its path loss values to the other active BS's. We consider a general model to characterize the relationship between the feasible rate and the SINR: the feasible rate of a user at location x served by BS n when the active set of BS's is $\mathcal{A}$ is given by:

$$C_{nx,\mathcal{A}} = f(\mathrm{SINR}_{nx,\mathcal{A}}), \tag{15}$$

where $f(\cdot)$ is some non-decreasing differentiable function. In the numerical experiments, we assume that this function corresponds to the Shannon formula, which provides a reasonable approximation of most real systems, up to a multiplicative constant:

$$C_{nx,\mathcal{A}} = W \log_2(1 + \mathrm{SINR}_{nx,\mathcal{A}}), \tag{16}$$

where W denotes the bandwidth. Denote by y_n the position of BS n. The SINR of a user in location x served by BS n while the set of active BS's in $\mathcal{A}$ is:

$$\mathrm{SINR}_{nx,\mathcal{A}} = \frac{P\Gamma(|x - y_n|)}{N_0 + P\sum_{m\in\mathcal{A}\setminus\{n\}} \Gamma(|x - y_m|)}, \tag{17}$$

where P denotes the common transmit power of the BS's, N_0 is the background noise level, and Γ is the path loss. We take values representative of 3G cellular networks: $P = 40$ dBm, $N_0 = -100$ dBm, $\Gamma(r) = -130 - 35\log_{10}(r)$ with r expressed in km (which corresponds to a path loss exponent equal to 3.5). Finally, we assume that the minimum distance between any user and any BS is strictly positive.

4.2 Traffic characteristics and network state

We consider a continuous setting for traffic characteristics, with an infinite number of flow classes. This arises when the feasible rate can take arbitrary values and when user locations can be anywhere in a continuous subset $\mathcal{X}$ of $\mathbb{R}^2$. The traffic model is then the following: users in an area of size dx around location x generate data flows of mean size σ_x according to a Poisson process of intensity $\lambda_x dx$. The traffic intensity generated in an area of size dx around location x is defined by $\rho_x dx = \lambda_x \times \sigma_x dx$. The total traffic intensity is denoted by $\rho = \int_{\mathcal{X}} \rho_x dx$, and the density of traffic generated around x by $p_x = \rho_x/\rho$ (so that $\int_{\mathcal{X}} p_x dx = 1$).

In the continuous setting, the network state $Z(t)$ is described by the number of active flows denoted by $Q(t)$, and by the vector $X(t) = (X_l(t), l = 1, \ldots, Q(t))$ describing the locations of the users corresponding to active flows, i.e., $X_l(t)$ is the class of the l-th active flow at time t.

Remark that considering a continuous setting instead of a discrete setting as in the previous sections does not make a fundamental difference, and the results derived in the discrete setting still hold.

The server assignment strategy corresponds to assigning users to BS's, termed *cell selection* in this context. Coordinated scheduling involves coordinating activity states of neighboring BS's with the aim of reducing inter-cell interference, and is referred to here as *inter-cell scheduling*.

4.3 Capacity with SA only

In this section we extend the results of Sect. 2 to characterize, in the continuous setting of traffic distribution, the optimal static cell selection scheme (for a given traffic distribution $(p_x)_{x\in\mathcal{X}}$), and then compute the network capacity. The case of non-interacting servers corresponds to constant inter-cell interference in this context, where the transmission rates are independent of the activity states of neighboring BS's. When there is server interaction, the transmission rates do depend on activity states of surrounding BS's, and this results in variable inter-cell interference.

Constant inter-cell interference We first focus on the case of constant inter-cell interference. Proposition 1 helps in determining an optimal static assignment strategy. However, it is not sufficient to completely characterize the network capacity, because in general there are many static assignments equalizing the cell loads. The following result characterizes the optimal static assignment strategy. In the continuous setting, an optimal assignment is defined by the boundaries between cells, i.e., for every given location x we have to determine which BS serves the users at this location. Denote by $\mathcal{B}^*_{n_1,n_2}$ the boundary between cells n_1 and n_2 in an optimal static assignment.

Proposition 9 *Consider a static assignment strategy achieving the maximum network capacity* C_{SA}. *For any pair of BS's* n_1, n_2, *there exists a non-negative* k *such that*:

$$\mathcal{B}^*_{n_1,n_2} \subseteq \left\{ x \in \mathcal{X} : \frac{C_{n_1 x}}{C_{n_2 x}} = k \right\}. \tag{18}$$

Proof The proof of this result involves swapping arguments. Consider a static assignment strategy stabilizing the network whenever possible. Then it equalizes the cell loads according to Proposition 1. Now consider two points x, y in $\mathcal{B}^*_{n_1,n_2}$ and assume for example that $\frac{C_{n_1 x}}{C_{n_2 x}} > \frac{C_{n_1 y}}{C_{n_2 y}}$. Then define a new static assignment obtained from the initial considered scheme after the following modifications: around location x in cell n_2 in the initial scheme, the users in an area of size such that the traffic intensity in that area is $\epsilon > 0$ (ϵ is chosen arbitrarily small) are served by BS n_1 instead of BS n_2. At location y, we make a modification to keep the load of cell n_2 constant: to this aim, around location y in cell n_1 in the initial scheme, the users in an area of size such that the traffic intensity in that area is $\epsilon' = \epsilon C_{n_2 y}/C_{n_2 x}$ (ϵ is arbitrarily small) are served by BS n_2 instead of BS n_1. The loads of all cells except that of cell n_1 remain unchanged. The load of cell n_1 is decreased by $-\epsilon(\frac{C_{n_2 x}}{C_{n_1 x}} - \frac{C_{n_2 y}}{C_{n_1 y}})/C_{n_2 x} > 0$. The new static assignment is strictly better than the initial scheme, in the sense that the cell loads are smaller than those with the initial assignment. This contradicts Proposition 1. □

Variable inter-cell interference We now examine the case of variable inter-cell interference. As observed in Bonald et al. (2004a), the model is quite complex due to the intricate correlations between the activity states of the various BS's. In Sect. 2.2 first- and second-degree bounds for the capacity were identified. In the following, we provide further results.

Equation (8) provides a lower bound for the capacity with server selection. However, determining the traffic distribution α^* leading to the greater second-degree bound

$$\hat{\mathsf{C}}_{\mathrm{SA}} = \max_{\alpha} \left(\max_{n\in\mathcal{N}} \sum_{x\in\mathcal{X}} \frac{\alpha_{x,n} p_x}{\overline{C}_{nx}(\alpha)} \right)^{-1}.$$

is not an easy task. For example, in general, it turns out impossible to characterize the *optimal* loads of the non-saturated cells: there is no analog of Proposition 1. However, in the continuous setting, first note that the optimal traffic distribution for the second-degree bound is such that all flows at a given location are served by the same BS. Then it remains to identify the boundaries between cells. These boundaries must satisfy the properties given in the following proposition. Denote by $\mathcal{B}^*_{n_1,n_2}$ the boundary between cells n_1 and n_2 (possibly empty). Denote by S the set of BS in this configuration that are always active. Without loss of generality we can assume that S reduces to a single cell, say cell 1.[1]

Proposition 10 *Consider a static assignment corresponding to the second-degree bound maximizing* $\hat{\mathsf{C}}_{\mathrm{SA}}$.

(i) *If* $n \neq 1$,

$$\mathcal{B}^*_{1,n} \subseteq \left\{ x \in \mathcal{X} : \frac{C_{nx,\mathcal{N}}}{\overline{C}_{1x}} = -\int_{\mathrm{cell}1} \frac{dy \hat{\mathsf{C}}_{\mathrm{SA}}}{\overline{C}^2_{1y}} \times \frac{\partial \overline{C}_{1y}}{\partial \rho_n} \right\}. \tag{19}$$

(ii) *If* $n_1, n_2 \neq 1$,

$$\mathcal{B}^*_{n_1,n_2} \subseteq \left\{ x \in \mathcal{X} : \int_{\mathrm{cell}1} \frac{dy \hat{\mathsf{C}}_{\mathrm{SA}}}{\overline{C}^2_{1y}} \times \left(\frac{1}{C_{n_1x,\mathcal{N}}} \frac{\partial \overline{C}_{1y}}{\partial \rho_{n_1}} - \frac{1}{C_{n_2x,\mathcal{N}}} \frac{\partial \overline{C}_{1y}}{\partial \rho_{n_2}} \right) = 0 \right\}. \tag{20}$$

Proof We only provide a proof of (i), since that of (ii) is similar. Assume for example that, for $n \neq 1$ and $x \in \mathcal{B}^*_{1,n}$,

$$\nu = \frac{C_{nx,\mathcal{N}}}{\overline{C}_{1x}} - \int_{\mathrm{cell}1} \frac{dy \hat{C}_{\mathrm{SA}}}{\overline{C}^2_{1y}} \times \frac{\partial \overline{C}_{1y}}{\partial \rho_n} > 0.$$

Starting from the initial network, we increase the load of cell n by ϵ (arbitrarily small) by serving users located around location x in cell 1 by BS n (instead of BS 1). A first-order Taylor expansion of the load of cell 1, initially equal to 1, gives that this load is decreased by $\epsilon\nu$. In the new network all cells have a load strictly less than 1, and hence the initial network does not correspond to the second-degree bound maximizing $\hat{\mathsf{C}}_{\mathrm{SA}}$, a contradiction. □

We now present the numerical experiments that we performed to corroborate the analytical findings.

4.4 Two-cell network

We first consider the two-cell network examined in earlier sections. We consider a continuous setting for the traffic distribution. In the examples provided below, heterogeneous traffic refers to a traffic intensity that is at a maximum at a distance of 0 from BS 1 and decreases linearly with no traffic at a distance of 0 from BS 2. Homogeneous traffic refers to a uniform traffic distribution between the two BS's. We will further consider two cases: two cells in isolation, i.e, no interferers, and two cells in an infinite linear network. For the latter case,

[1] Indeed, S contains several BS's when the network has some symmetry properties. In that case, we can break the symmetry and build a sequence of 'asymmetric' networks (i.e., such that $S = \{1\}$) converging to the initial network.

the capacity is sensitive in general, and we present results assuming constant interference from all other BS's.

Figure 1 plots the network capacity for increasing cell radius for homogeneous and heterogeneous loads in two cells in isolation, and for different resource allocation strategies. When the resource allocation is not coordinated (the curve 'Nothing'), we assume that each user is served by the closest BS irrespective of the cell loads. The capacities under inter-cell scheduling and cell selection are given by Proposition 7 and (4), respectively. Note that we have not included the curve for the resource allocation using both inter-cell scheduling and cell selection in order to keep the figures uncluttered. The network capacity under this combined policy is quite close to that of the higher of the two single policies in general. For small cells with both types of traffic distribution, inter-cell scheduling provides significant gains. For very dense networks, cell selection also results in significant capacity gains compared to no policy and is sightly worse than inter-cell scheduling. However since cell selection may be relatively easier to implement, with inter-cell scheduling requiring coordination of BS's, cell selection may be seen an immediate way to gain more capacity in very dense networks. As expected, when cells are larger and there is less interference, cell selection results in higher network capacity than inter-cell scheduling and this is more evident when the traffic distribution is heterogeneous. For such noise-limited networks, cell selection tends to balance the loads among BS's. For dense interference-limited networks this is not the case, as shown in Fig. 2. In fact, when the cells are very small, the capacity is maximized when BS 1 carries all the traffic, leading to an idle BS 2 and thus strongly unbalanced loads.

Figure 3 plots the network capacity against cell radius for two cells in an infinite linear network. The additional interference here means that for dense networks the gains in capacity due inter-cell scheduling are smaller and even more so for cell selection. It remains the case however, that for sparse networks cell selection is the best approach.

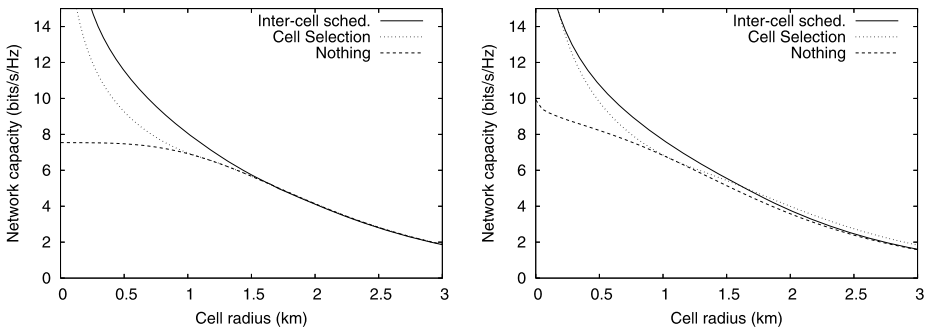

Fig. 1 Network capacity: 2 cells in isolation, (*left*) homogeneous traffic, (*right*) heterogeneous traffic

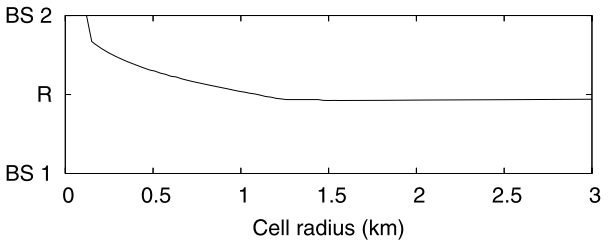

Fig. 2 Optimal cell selection scheme: cell boundary, 2 cells in isolation, and heterogeneous traffic

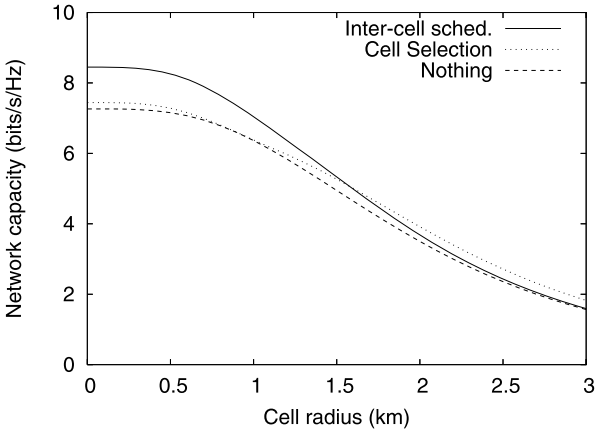

Fig. 3 Network capacity: 2 cells in an infinite linear network, heterogeneous traffic

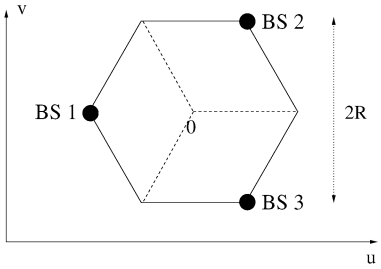

Fig. 4 A 3-cell network

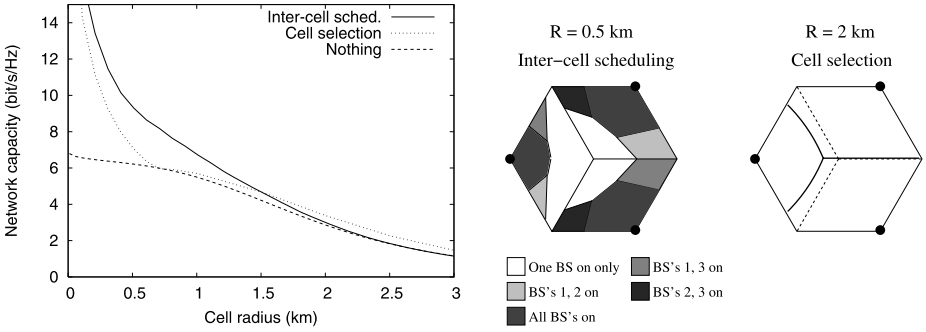

Fig. 5 (*Left*) Capacities under various resource allocation strategies, and (*right*) the optimal inter-cell scheduling scheme and static cell selection scheme (leading to the greatest second degree bound)

4.5 Three facing sectors

We now consider a network of three BS's as represented in Fig. 4. We assume that the traffic has a spatial distribution such that the traffic intensity generated at position $x = (u, v)$ is proportional to $\frac{2R}{\sqrt{3}} - u$ (more traffic is generated in Cell 1 than in Cell 2 or 3). In this network, Cells 2 and 3 are equivalent which simplifies the analysis.

The capacity with cell selection only is known to be sensitive to the flow size distribution. In Fig. 5, the capacities without resource allocation coordination or with cell selection

only are computed applying the second-degree bound. Here we present the second-degree bounds (7) and (8). In case of cell selection, to find the cell boundaries corresponding to the network whose capacity second degree bound is maximum, we apply results of Proposition 10. Figure 5 shows these boundaries when $R = 2$ km, and in this case the optimal static cell selection almost equalizes the cell loads).

Finding the optimal static inter-cell scheduling in a heterogeneous network is challenging, but it can be characterized identifying the region of the cell where users are served in a given profile. This has been done in Liu and Virtamo (2006).

As expected the capacity gain applying cell selection or inter-cell scheduling can be very high when the cell radius is very small: in that case, the optimal cell selection consists in serving all traffic by BS 1 and the optimal inter-cell scheduling in using transmission profiles where only one BS is active. When the cell radius is large, inter-cell scheduling is ineffective, but cell selection, which equalizes the cell loads, still shows significant capacity gains.

5 Conclusion

We have investigated the stability of wireless data networks with interfering base stations supporting spatially distributed users with finite random service demands. It was shown how these networks may be modeled as interacting queues whose service rates not only depend on the user class, but also on the set of active servers. We have used the stability results to examine the potential capacity gains from various types of resource allocation strategies, e.g. dynamic server assignment, coordinated scheduling, or a combination of these two.

Several natural avenues for further research present themselves. First of all, it would be interesting to construct dynamic server assignment schemes that only use queue length information, and do not rely on any knowledge of the workloads. In many situations, queue length information can be easily obtained or implicitly conveyed to users which selfishly select servers based on perceived throughput under fair resource sharing disciplines, whereas the workload involves knowledge of remaining service requirements which may not be readily available. Second, the stability may depend on the scheduling disciplines employed by the individual servers, in particular the value of κ in case of κ-fair scheduling. It would be interesting to explore what values of κ, possibly as function of the set of active servers, provide maximum stability.

References

Bender, P., Black, P., Grob, M., Padovani, R., Sindhushayana, N., & Viterbi, A. (2000). CDMA/HDR: a bandwidth-efficient high-speed wireless data service for nomadic users. *IEEE Communications Magazine*, *38*(7), 70–77.

Bonald, T., & Proutière, A. (2003). Wireless downlink channels: user performance and cell dimensioning, In *Proc. ACM Mobicom*.

Bonald, T., Borst, S. C., Hegde, N., & Proutière, A. (2004a). Wireless data performance in multi-cell scenarios. In *Proc. ACM sigmetrics/performance 2004* (pp. 378–388).

Bonald, T., Borst, S. C., & Proutière, A. (2004b). How mobility impacts the flow-level performance of wireless data networks. In *Proc. IEEE infocom*.

Bonald, T., Jonckheere, M., & Proutière, A. (2004c). Insensitive load balancing. In *Proc. ACM sigmetrics/performance 2004* (pp. 367–377).

Bonald, T., Borst, S. C., & Proutière, A. (2005). Inter-cell scheduling in wireless data networks. In *Proc. European wireless conf.*

Bonald, T., Massoulié, L., Proutière, A., & Virtamo, J. (2006). A queueing analysis of max-min fairness, proportional fairness and balanced fairness. *Queueing Systems*, *53*, 65–84.

Chernova, N., & Foss, S. (1998). On the stability of a partially accessible multi-station queue with state-dependent routing. *Queueing Systems*, *29*(1), 55–74.

Cohen, J. W. (1984). On a functional relation in three complex variables; three coupled processors. Technical Report 359, Mathematical Institute, University of Utrecht.

Cohen, J. W., & Boxma, O. J. (1983). *Boundary value problems in queueing system analysis*. Amsterdam: North-Holland.

Dai, J. G. (1995). On positive Harris recurrence of multiclass queueing networks: A unified approach via fluid limit models. *Annals Applied Probability*, *5*, 49–77.

Das, S., Viswanathan, H., & Rittenhouse, G. (2003). Dynamic load balancing through coordinated scheduling in packet data systems. In *Proc. IEEE infocom*.

Fayolle, G., & Iasnogorodski, R. (1979). Two coupled processors: the reduction to a Riemann-Hilbert problem. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, *47*, 325–351.

Foley, R. D., & McDonald, D. (2001). Join the shortest queue: stability and exact asymptotics. *Annals of Applied Probability*, *11*, 569–707.

Liu, S., & Virtamo, J. (2006). Inter-cell coordination with inhomogeneous traffic distribution. In *Proc. second EuroNGI conf. next generation Internet design and engineering, NGI 2006*, Valencia, Spain (pp. 64–71).

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

Sang, A., Wang, X., Madihian, M., & Gitlin, R. D. (2004). Coordinated load balancing, hand off cell-site selection, and scheduling in multi-cell packet data systems. In *Proc. ACM Mobicom*.

Squillante, M., Xia, C. H., Yao, D., & Zhang, L. (2001). Threshold-based priority policies for parallel-server systems with affinity scheduling. In *Proc. IEEE American control conf.* (pp. 2992–2999).

Stolyar, A. L. (2005). Optimal routing in output-queued flexible server systems. *Probability in the Engineering and Informational Sciences*, *19*, 141–189.

Tassiulas, L., & Ephremides, A. (1992). Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multi-hop radio networks. *IEEE Transactions on Automatic Control*, *37*, 1936–1938.