Queueing Systems (2019) 93:227–268
https://doi.org/10.1007/s11134-019-09633-y

Job assignment in large-scale service systems with affinity relations

Ellen Cardinaels[1] · **Sem C. Borst**[1] · **Johan S. H. van Leeuwaarden**[1]

Received: 21 December 2018 / Revised: 15 September 2019 / Published online: 10 October 2019

Abstract
We consider load balancing in service systems with affinity relations between jobs and servers. Specifically, an arriving job can be assigned to a fast, primary server from a particular selection associated with this job or to a secondary server to be processed at a slower rate. Such job–server affinity relations can model network topologies based on geographical proximity, or data locality in cloud scenarios. We introduce load balancing schemes that assign jobs to primary servers if available, and otherwise to secondary servers. A novel coupling construction is developed to obtain stability conditions and performance bounds. We also conduct a fluid limit analysis for symmetric model instances, which reveals a delicate interplay between the model parameters and load balancing performance.

Keywords Load balancing · Stochastic coupling · Fluid limit · Job scheduling · Network topology

Mathematics Subject Classification 60K25 · 68M20 · 90B15 · 90B22 · 90B35

1 Introduction

Load balancing algorithms play a crucial role in distributing jobs among multiple servers and have attracted strong renewed interest due to proliferation of large data centers and cloud computing. Well-known load balancing algorithms include, for instance, the Join-the-Shortest-Queue (JSQ), Join-the-Shortest-Queue-d (JSQ(d)) and Join-the-Idle-Queue (JIQ) policies. These policies have been extensively analyzed in an overarching framework called the *supermarket model*. This framework consists of a single dispatcher that immediately routes the arriving jobs to one of the N identical

✉ Ellen Cardinaels
e.cardinaels@tue.nl

[1] Department of Mathematics and Computer Science, Eindhoven University of Technology, P.O. Box 513, 5600 MB Eindhoven, The Netherlands

parallel servers according to the assignment policy under consideration. The servers are identical in the sense that each of them can handle any arriving job and all servers process jobs at the same rate. The JSQ policy assigns each arriving job to the server with the shortest queue length and has strong stochastic optimality properties among the class of policies without advance knowledge about the service requirements [5,28]. The JSQ policy involves a significant communication burden, however, which may be prohibitive in large systems.

This scalability issue has spurred an interest in the JSQ(d) policy which assigns a job to the server with the shortest queue length among $d \geq 2$ randomly selected servers. Mitzenmacher [18] and Vvedenskaya et al. [27] analyzed the JSQ(d) policy in an asymptotic regime where the total arrival rate and the number of servers grow proportionately larger. Substantial performance gains were established compared to purely random assignment, even for $d = 2$. Mukherjee et al. [19] show that the waiting time of an arriving job in fact vanishes when d tends to infinity as the number of servers grows large. A vanishing waiting time is also achieved by the JIQ policy which directs arriving jobs to an idle server or a randomly selected server if all servers are occupied [16]. The JIQ policy only has a constant communication overhead per job, but requires memory at the dispatcher. We refer to Van der Boor et al. [26] and Gamarnik et al. [8] for further details.

As mentioned above, a key feature of the supermarket framework is the exchangeability of the servers in the sense that any job can be handled equally well by any server, which is often not the case in practice. In the present paper, we will focus on a scenario where jobs or servers are not intrinsically different, but where particular servers might be better equipped to process certain jobs because of affinity or compatibility relations. We refer to the model as the *affinity-scheduling model*. Such affinity relations may, for example, arise due to geographical proximity in spatial settings, or data locality in content distribution or transaction processing applications.

More concretely, once a job arrives, there is a subset of the servers (referred to as the primary selection) that can process it at rate μ_1. However, it might be beneficial to assign this job to one of the remaining servers, if this allows an immediate start of the service. Obtaining service at a server that was not in the primary selection comes with a price; the service rate μ_2 will be smaller than μ_1. In a recent paper by Gardner et al. [9], a similar assignment policy is considered for a loosely related but different model where servers are heterogeneous. The jobs are statistically identical in this setting, i.e., a specific subset of servers is faster for all jobs. In Sect. 2, the assignment and scheduling policy of the affinity-scheduling model will be described in greater detail as well as the similarity and difference with the setup in [9].

One particular model instance is the *neighborhood model*, where the geographical proximity of the underlying network has been taken into account. Assume that the service system is built on top of a graph structure G_N, then the primary selection for a job consists of the server where it arrives and the immediate neighbors of this server determined by G_N. The neighborhood model extends the models constructed by Gast [10], Turner [25] and Mukherjee et al. [19]. In these settings, it is assumed that all nodes have equal arrival rates and jobs can only be forwarded to direct neighbors; it is not possible to redirect an arriving job to any other nodes. The model constructed by

Yekkehkhany et al. [29] does allow for jobs to be redirected to higher-degree neighbors to be served at lower rates.

Another model instance that we will investigate is the *combinatorial model*. Let d be a fixed integer, then any selection of d servers could occur as a primary selection for an arriving job. In addition, the arrival rates will be equal among the server selections which strengthens the symmetric nature of the combinatorial model.

The lack of exchangeability among the servers makes the affinity-scheduling model complicated to analyze in general. The analytical techniques that are most commonly used in the context of the supermarket model, such as mean field limits and even standard coupling arguments, fundamentally rely on this exchangeability. These techniques can only be applied for the combinatorial model. For the general model, and in particular the neighborhood model, the investigation of load balancing issues is challenging and enters largely uncharted methodological territory.

We will establish a stochastic dominance result for the occupancy process of the general affinity-scheduling model, which will yield a sufficient stability constraint as an immediate by-product. Exploiting the coupling of this dominance result, we can derive two tighter dominance results for the neighborhood model, which will in particular hold if the underlying graph structure is rather dense. To the best of our knowledge, these are the first results that explicitly capture the impact of network structure on load balancing performance.

For the combinatorial model, we will conduct a fluid limit analysis. A trajectory of the fluid limit will converge to one of the possibly multiple fixed points, depending on the mutual relationships of the model parameters and the initial configuration of the system. When the fixed point is unique, we demonstrate that this provides a good approximation for the intractable stationary distribution in a finite server setting. When multiple fixed points occur, we observe the phenomenon of 'tunneling' described by [11]. The stochastic process will switch between multiple modes corresponding to the locally stable fixed points of the fluid limit.

The remainder of this paper is organized as follows: A detailed model description is provided in Sect. 2. Next, the main stochastic dominance result and its associated coupling are presented in Sect. 3, as well as two tighter results for specific instances of the neighborhood model. In Sect. 4, we present a fluid limit analysis of the affinity-scheduling policy for combinatorial models. The proofs of all results are deferred to Sect. 5. Finally, Sect. 6 provides concluding remarks and some directions for further research.

2 Model description

We now describe the affinity-scheduling model with N servers. Let $\mathcal{P}(\{1,\ldots,N\})$ denote the power set of all servers. For a selection $S \in \mathcal{P}(\{1,\ldots,N\})$, jobs arrive as a Poisson process with rate $\lambda_S \geq 0$. For these jobs, the servers in S and S^c are called the primary and secondary servers, respectively. An arriving job can be assigned as a type-I job to a primary server or as a type-II job to a secondary server. Type-I jobs have independent and exponentially distributed service times with parameter $\mu_1 > 0$ and are favored by the server over the type-II jobs. Type-II jobs have on average longer

service times which are independent and exponentially distributed with parameter μ_2 ($0 < \mu_2 < \mu_1$). It is important to note that the job type is not predetermined on arrival, but established by the assignment policy. The main idea behind our assignment policy is: 'Assign a job to a server in the primary selection unless it might be beneficial to assign it to a secondary server even though the service time might be longer.' The rationale for this is to reduce the waiting time of a job. More precisely, the *assignment policy* goes through the following three steps:

1. If there is at least one completely idle server in the primary selection S, then assign the arriving job as a type-I job to one of these servers.
2. Otherwise, if there is at least one completely idle server in the secondary selection S^c, then assign the arriving job as a type-II job to one of these servers.
3. If there are no idle servers present, then assign the job as a type-I job to the primary server with the smallest number of type-I jobs. Ties are broken according to the number of type-II jobs, in favor of a lower number.

When the second step is omitted, our policy resembles a JSQ($|S|$) policy with $|S|$ the cardinality of the primary selection. However, the cardinality of the server selection is allowed to differ among arriving jobs in our model and the server selection S itself is not sampled uniformly at random as is the case in a JSQ($|S|$) policy. Moreover, the second step can be interpreted as a JIQ policy on the set of secondary servers. Our affinity-scheduling policy thus shares similarities with both policies.

Denote the configuration of a server, i.e., the number of type-I and type-II jobs in its queue, by (i, j), $i \geq 0$ and $j \in \{0, 1\}$. As an illustrative example of the assignment policy, suppose, for a given arriving job, the primary and secondary servers have $\{(1, 0), (1, 1), (1, 1), (4, 0)\}$ and $\{(1, 0), (1, 1), (1, 1), (3, 1)\}$ as their configurations, respectively. Under the assignment policy, the third step will be applied and the primary server with configuration $(1, 0)$ will receive an additional type-I job. Notice that under this strategy, an arriving job will never be assigned as a type-II job to a server that already has a job in its queue. This implies that if the initial configuration has at most one type-II job, the number of type-II jobs per server will never exceed one. Without essential loss of generality, we will assume that the initial configuration indeed satisfies this constraint. Furthermore, we assume that the routing decision time and transit times are negligible.

There is no lower bound imposed on the value of μ_2, which strengthens the fact that we focus primarily on the type-I jobs. In general, type-I jobs are the preferred type of jobs, which also manifests itself in the *scheduling policy*. Each server operates under a preemptive priority scheduling discipline in favor of the type-I jobs. Moreover, type-I jobs are served in order of arrival.

Let $N_{n,j}(t)$ denote the number of type-j jobs at server $n \in \{1, \ldots, N\}$ at time t. The configuration of server n is then given by $\big(N_{n,\mathrm{I}}(t), N_{n,\mathrm{II}}(t)\big) \in \mathbb{N} \times \{0, 1\}$ with state space $(\mathbb{N} \times \{0, 1\})^N$. The vector $\big(N_{n,\mathrm{I}}(t), N_{n,\mathrm{II}}(t)\big)_n$ evolves as an irreducible, time-homogeneous Markov process. However, the length of this vector grows with the number of servers which is therefore not well suited for, for instance, a fluid limit analysis. Consequently, we introduce different variables that are more server centric and will be more convenient in proving stochastic dominance and analyzing the fluid limit. Define $Q_{ij}(t)$ as the number of servers with i type-I jobs and j type-II jobs at

time t, with $i \geq 0$ and $j \in \{0, 1\}$. Then,

$$\overline{Q}_{ij}(t) \doteq \sum_{k \geq i} Q_{kj}(t) \tag{1}$$

denotes the number of servers with at least i type-I jobs and exactly j type-II jobs. We note that $\overline{Q}^N_{00}(t) + \overline{Q}^N_{01}(t) = N$ by definition. Since the stochastic dominance result in the next section will focus on the type-I jobs, we also introduce the following variables:

$$\overline{Q}_{m+}(t) \doteq \sum_{i=m}^{\infty} \overline{Q}_i(t), \tag{2}$$

where $\overline{Q}_i(t)$ denotes the number of servers with at least i type-I jobs, i.e., $\overline{Q}_i(t) = \overline{Q}_{i0}(t) + \overline{Q}_{i1}(t)$. It is important to note that these variables will no longer lead to a Markov process representation in the general settings mentioned in the introduction. This immediately limits the number of available techniques to analyze the performance.

Let $\mathcal{S}$ denote the subset of $\mathcal{P}(\{1, \ldots, N\})$ with strictly positive arrival rates. Besides the general setting where $\mathcal{S}$ can be any subset of $\mathcal{P}(\{1, \ldots, N\})$, we will also investigate some more restricted cases. In the *neighborhood model* on the graph topology G_N, each node represents a server and the edges represent underlying relations between them. Then each selection in $\mathcal{S}$ consists of a server and its neighbors determined by G_N. In total, $\mathcal{S}$ contains N different server selections and jobs arrive to each of them independently at a uniform rate $\lambda > 0$. This model instance captures the situation where a job's physical arrival location plays a role in its service time at the various servers.

Let $\mathcal{S}$ consist of all possible server selections of size d. The cardinality of $\mathcal{S}$ is $\binom{N}{d}$, and henceforth, we refer to this model as the *combinatorial model*. We assume a uniform arrival rate ν per selection. We let $\nu = \lambda N / \binom{N}{d}$ per selection so that the total rate is given by λN. Observe that the combinatorial model captures the situation where a selection of d servers is drawn uniformly at random as the primary selection for each job.

Remark 1 There are also instances of the affinity-scheduling model that are not captured by either the neighborhood model or the combinatorial model. On the one hand, there are also model instances with non-uniform arrival rates $(\lambda_S)_S$ per server selection. On the other hand, some model instances could be intermediate versions of the neighborhood model and the combinatorial model. As an example, suppose a job arrives to a primary selection that consists of the servers $1, \ldots, 5$ or an arbitrary selection of size two of the remaining servers. Then, $\mathcal{S}$ consists of $\{1, \ldots, 5\}$ and all pairs of servers of $6, \ldots, N$.

As mentioned in introduction, Gardner et al. focus on a model with heterogeneous servers which is loosely related to ours. The system under consideration consists of exactly k_F servers operating at rate μ_1 (fast servers) and k_S servers operating at rate $\mu_2 < \mu_1$ (slow servers) [9]. For each arriving job, a primary selection of size d_F

is uniformly selected from the fast servers and a secondary selection of size d_S is selected from the slow servers. Then, an allocation strategy similar to ours, referred to as JSQ(d_F, d_S), is applied with the only difference that the second and third steps are not always applied even if the conditions are fulfilled. For instance, even if the condition in step two is fulfilled, with probability $1-p_S$ the arriving job will still be forwarded to the (busy) fast server with the shortest queue length. Similarly, the third step is only applied with probability p_F once its condition is fulfilled and there are no idle severs in both selections. So, the major difference between the two models is the way heterogeneity between the servers manifests itself. Both models coincide in the very special case when the allocation strategy in [9] utilizes the model parameters $d_F = k_F$, $d_S = k_S$ and $p_F = p_S = 1$ and the set of server selections $\mathcal{S}$ in the affinity-scheduling model consists of only one set of servers that coincides with the fast servers in [9].

3 Stochastic dominance and coupling

In this section, we establish several stochastic dominance results for our affinity-scheduling strategy. We will construct a stochastic coupling that allows a comparison with various reference systems in terms of the ordered server states, and refer to this coupling as the *restructure coupling*, or shorter *R-coupling*. In contrast to the affinity system, the various reference systems all involve N exchangeable servers and are therefore far more amenable to (asymptotic) analysis, yielding tractable performance bounds. So, the coupling builds a bridge between a structured system on the one hand and an unstructured system on the other hand, hence the name of the coupling. From the previous section, we know that there is at most one type-II job present. Due to this boundedness of the number of type-II jobs, we will only count the number of type-I jobs in the affinity system. One reason for this is that only these jobs could possibly lead to unstable behavior.

3.1 General framework

Before we elaborate on the specific results, the general framework of the *R-coupling* is presented. The common features and differences of the specific generalizations of the coupling for different model instances are exposed.

While each of the N servers in the reference system processes jobs in a FCFS manner at rate μ_1, the various specific incarnations differ in the value of the normalized arrival rate per server λ_0 and the policy for assigning jobs. The choice of the specific reference system is aligned with the properties of the affinity system in terms of the server selections $\mathcal{S}$ and the associated arrival rates λ_S, $S \in \mathcal{S}$. Loosely speaking, we obtain increasingly tight dominance results under increasingly restrictive symmetry and structural conditions on the server selections $\mathcal{S}$ and the associated arrival rates. The three specific variants for the reference system that we consider operate under either (i) a purely random assignment (RA) policy, (ii) a MJSQ(k) policy (as specified later) or (iii) a JSQ(k) policy (as described in the introduction). While the RA system

provides exact upper bounds in terms of independent M/M/1 queues, the MJSQ(k) and JSQ(k) systems yield asymptotic upper bounds based on fluid limits.

The dominance results revolve around stochastic majorization properties in terms of the ordered server states. Specifically, define $\overline{Q}^{\text{aff}}_{m+}(t)$ and $\overline{Q}^{\text{ref}}_{m+}(t)$ as the variables in (2) for the affinity and reference system, respectively. We will establish results of the form

$$\{(\overline{Q}^{\text{aff}}_{m+}(t))_{m\geq 1}\}_{t\geq 0} \leq_{\text{st}} \{(\overline{Q}^{\text{ref}}_{m+}(t))_{m\geq 1}\}_{t\geq 0}.$$

This majorization result indicates that the number of type-I jobs residing in the mth or higher-ordered queue position in the affinity system is stochastically bounded from above by the number of jobs residing in the mth or higher-ordered queue position in the reference system. In particular, taking $m = 1$, this implies that the total number of type-I jobs in the affinity system is stochastically bounded from above by the total number of jobs in the reference system. As noted earlier, we know the exact distribution of the latter quantity in the RA system and have an asymptotic result for the MJSQ(k) and JSQ(k) systems.

In order to prove the stochastic majorization properties, we introduce the R-coupling to construct sample paths for the affinity and reference systems on a joint probability space for which the stated inequalities hold in a deterministic way [15,22,24]. The servers in both systems can be ordered in an ascending way according to number of (type-I) jobs in their queue. Let $N_{[n],\text{I}}(t)$ and $N_{[n]}(t)$ denote the number of type-I jobs in the affinity system and in the reference system at the nth ordered server at time t, respectively. For all three specific reference systems, the common proof concept is to ensure that under the coupling the two following key properties always hold with respect to the ordered server states as illustrated in Fig. 1. Let t indicate the moment that an event occurs in the coupled systems.

(a) If $N_{[n],\text{I}}(t) = N_{[n],\text{I}}(t^-) + 1$, then $N_{[\tilde{n}]}(t) = N_{[\tilde{n}]}(t^-) + 1$ with $\tilde{n} \in \{n, \ldots, N\}$.
So, an arrival of a type-I job in the affinity system must give rise to an arrival at a higher-ordered server in the reference system.

(b) If $N_{[n]}(t) = N_{[n]}(t^-) - 1$, then $N_{[n],\text{I}}(t) = \max(N_{[n],\text{I}}(t^-) - 1, 0)$.
So, a service completion in the reference system must force a service completion of a type-I job at the same ordered server in the affinity system (unless there is no type-I job at this server).

We can prove the following general lemma.

Lemma 1 (R-coupling) *If a stochastic coupling between the affinity system and the reference system can be constructed such that (a) and (b) are satisfied, then* $(\overline{Q}^{\text{aff}}_i(t))_{i\geq 1}$ *is majorized by* $(\overline{Q}^{\text{ref}}_i(t))_{i\geq 1}$ *for* $t \geq 0$*, that is,*

$$\sum_{i=m}^{\infty} \overline{Q}^{\text{aff}}_i(t) \leq \sum_{i=m}^{\infty} \overline{Q}^{\text{ref}}_i(t) \tag{3}$$

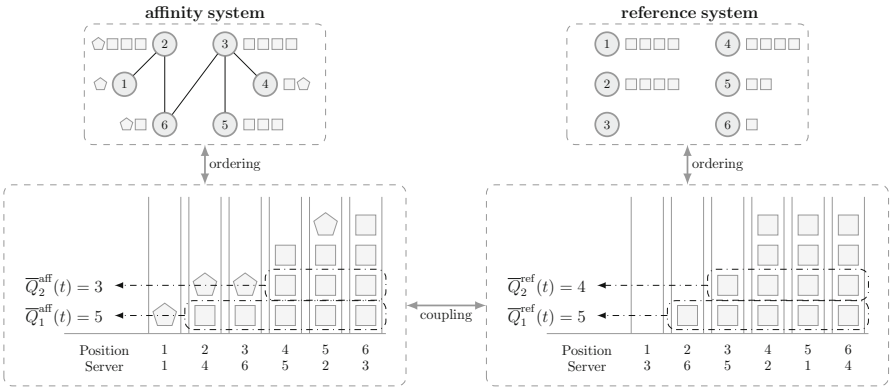


Fig. 1 A schematic representation of the two ordered systems with $N = 6$ servers at time t. The affinity system, operating under the affinity-scheduling policy, consists of type-I (squares) and type-II (pentagons) jobs. The N exchangeable servers in the reference system make no distinction between the jobs

for all $m \geq 1$, *provided that the initial configurations of both systems satisfy this inequality.*

More precisely, the vector $(\overline{Q}_i^{\text{ref}}(t))_{i\geq 1}$ weakly submajorizes the vector $(\overline{Q}_i^{\text{aff}}(t))_{i\geq 1}$. If for $m = 1$ the inequality in (3) holds with equality, then this result implies that the affinity-scheduling policy disperses the type-I jobs more evenly among the servers than the considered assignment policy in the reference system. The proof of Lemma 1 is discussed in Sect. 5.1. In the remainder of this section, we will precisely describe the affinity coupling for each of the reference systems under consideration and verify that the properties (a) and (b) are satisfied. The coupling at service completion epochs to ensure property (b) as further detailed below is fairly standard and common across all three reference systems. The coupling at arrival epochs depends on the assignment policy under consideration in the reference system. Although the general framework is highlighted below, the precise realizations will be illustrated in the following subsections.

Coupling at arrival epochs In contrast to the service completions, the coupling at arrival epochs to guarantee property (a) is novel and highly specific to the reference system under consideration. Due to the lack of exchangeability among servers, the coupling at arrival epochs involves a further subtle complication that does not arise in constructing sample path comparisons in the context of the ordinary supermarket model. Even though we compare the evolution of the two systems in terms of the $\overline{Q}_i$ variables as usual, these generally do not provide a Markovian state description for the affinity system as noted earlier in Sect. 2. In particular, the transitions at arrival epochs intricately depend on the server selections $\mathcal{S}$ and cannot be suitably represented in terms of the $\overline{Q}_i$ variables.

Coupling at service completion epochs The coupling generates potential service completions at rate $\mu_1 N$, but the aggregate service rate in either the affinity or the reference system might be lower than $\mu_1 N$ because of servers being idle or only working at rate μ_2 on type-II jobs. Let P_{aff} and P_{ref} be the sets of ordered positions of

servers in the affinity and reference system, respectively, that are working on (type-I) jobs just before some time t at which a potential service completion occurs. Define P as the intersection $P_{\text{aff}} \cap P_{\text{ref}}$ which equals P_{aff} or P_{ref} due to the ordering and the preemptive strategy of the affinity-scheduling policy. A random variable X_t, drawn from a uniform distribution on $[0, 1]$, decides which of the following actions is selected:

(i) $0 \leq X_t \leq \frac{|P|}{N}$: Sample uniformly at random a position n from P; a departure will take place at time t in both systems at the nth ordered server.
(ii) $\frac{|P|}{N} < X_t \leq \frac{|P_\alpha|}{N}$ where α is 'aff' or 'ref': Sample uniformly at random a server position from $P_\alpha \setminus P$; one job will be removed from the corresponding server in system α at time t.
(iii) $X_t > \frac{\max\{|P_{\text{aff}}|,|P_{\text{ref}}|\}}{N}$: No real departure will occur among the type-I jobs in the affinity system or the jobs in the reference system.

We note that the total departure rate of type-I jobs from the affinity system is indeed given by $\mu_1|P_{\text{aff}}|$, likewise for the reference system with a total departure rate of $\mu_1|P_{\text{ref}}|$. The idea to work with intersections of the active server sets comes from [20, Section 4].

3.2 R-coupling with the general model

We now consider a general structure for the server selections $\mathcal{S}$ and the corresponding arrival rates $\{\lambda_S \mid S \in \mathcal{S}\}$ per server selection. The reference system will operate under the RA policy with arrival rate λ_0 per server. Thus, $\lambda_0 < \mu_1$ is a sufficient stability condition for the reference system. So the purpose of this subsection is twofold: the affinity coupling is illustrated in the general setting of our affinity-scheduling policy in order to obtain a stochastic dominance result, and a stability condition is obtained as an immediate by-product.

The choice of λ_0 is determined by the arrival rates per server selection in the affinity system, namely

$$\lambda_0 \doteq \min_{p_{Sn}} \left\{ \max_n \left\{ \lambda_n^* = \sum_{S \in \mathcal{S}: n \in S} \lambda_S p_{Sn} \;\middle|\; \sum_{n \in S} p_{Sn} = 1 \text{ with } p_{Sn} \geq 0, \forall n \in S \right\} \right\}. \tag{4}$$

In order to achieve a meaningful upper bound, we chose λ_0 as small as possible. Since we only count the number of type-I jobs in the affinity system, λ_0 must also be large enough to cope with 'worst-case scenarios' where no arriving job is assigned as a type-II job. Therefore, the optimization problem in (4) computes a solution that distributes the arriving jobs as type-I jobs among the servers as uniformly as possible. The variable p_{Sn} may be interpreted as the fraction of jobs with server selection S that are assigned to server $n \in S$. With this interpretation in mind, it is easily seen that at least one server must handle an arrival rate of λ_0 or larger in case jobs are only allowed to be executed as type-I jobs. Thus, $\lambda_0 < \mu_1$ is clearly a necessary stability condition for any policy in this case. The condition is sufficient as well, for instance for a simple static strategy that assigns a job with server selection S to server n with probability p_{Sn}. However, the implementation of this policy would require full knowledge of the

arrival rates λ_S. We will establish that the condition is also sufficient for the stability of our affinity-scheduling strategy, which does not rely on any knowledge of the arrival rates λ_S at all.

We now specify the R-coupling for the reference system with the RA policy.

Coupling at arrival epochs The coupling generates potential arrival events at rate $\lambda_0 N$. If a potential arrival occurs at time t, a position n^* from the set $\{1, \ldots, N\}$ is selected uniformly at random. For brevity, we simply refer to the server at the ordered position n^* as server n^*. An addition of a new job in the reference system will take place at this server n^*. Since this position was randomly selected, the coupling strategy will give rise to an addition according to the RA policy in a system with arrival rate λ_0 per server.

In order to determine whether an arrival event of a type-I job takes place in the affinity system and at which server this will happen, we follow the strategy described below. Two random variables, $Y_{t,1}$ and $Y_{t,2}$, are sampled from a uniform distribution on [0, 1] to take into account that the total arrival rate in the affinity system might be smaller than $\lambda_0 N$ and to select a server selection S for an arriving job. To make the decisions, we rely on the variables $(p^*_{Sn})_{S,n}$ that attain the minimum in (4). First, $Y_{t,1}$ establishes if an arrival occurs to a primary selection containing server n^*, which happens with probability $\lambda^*_{n^*}/\lambda_0$. If an arrival will take place, then a server selection S containing n^* is selected as the primary selection with probability $\lambda_S p^*_{Sn^*}/\lambda^*_{n^*}$ for which $Y_{t,2}$ is used. All remaining servers form the secondary selection. Note that the total arrival rate to a server selection S,

$$\lambda_0 N \sum_{n \in S} \frac{1}{N} \frac{\lambda^*_n}{\lambda_0} \frac{\lambda_S p^*_{Sn}}{\lambda^*_n}, \tag{5}$$

will indeed be equal to λ_S in the affinity system as $\sum_{n \in S} p^*_{Sn} = 1$ by definition. So, this approach will coincide with the arrival process of the general model described in Sect. 2. Once these selections are set for an arriving job, the assignment policy is applied as defined in Sect. 2. Due to the general structure of $\mathcal{S}$, it is not possible to determine the exact server at which a job is assigned in terms of the variables $(\overline{Q}_{ij})_{i,j}$. However, if the new job is assigned as a type-I job in the affinity system to one of the servers in S, it is known that the position of this server will be at most n^*. Since the newly arrived job in the reference system is assigned to server n^*, property (a) of the coupling is maintained.

We can state the following theorem that will lead to a sufficient stability condition. This theorem follows from the majorization result in Lemma 1 since the above-described coupling satisfies the general framework of the R-coupling.

Theorem 1 (General affinity-scheduling model) *Let λ_0, as defined in (4), be the arrival rate per server in the reference system operating under the* RA *policy. Then, for suitable initial conditions,*

$$\{(\overline{Q}^{\text{aff}}_{m+}(t))_{m \geq 1}\}_{t \geq 0} \leq_{\text{st}} \{(\overline{Q}^{\text{RA}}_{m+}(t))_{m \geq 1}\}_{t \geq 0} \tag{6}$$

holds for the general affinity-scheduling model with N servers.

Theorem 1 provides a stochastic upper bound for the total number of type-I jobs in the affinity system in terms of the number of jobs in a reference system with the RA policy by taking $m = 1$. Although this upper bound is sufficient to guarantee stochastic stability for $\lambda_0 < \mu_1$, we will develop tighter majorization results for particular instances of the neighborhood model in the next subsection.

It can be shown that the condition $\lambda_0 < \mu_1$ indeed implies the necessary and sufficient stability condition for a broader class of flexible parallel server systems in Harrison and López [12] and Stolyar [23]. In the setting of the affinity-scheduling model without type-II jobs, i.e., $\mu_2 = 0$, the condition $\lambda_0 < \mu_1$ is also necessary for stability. This scenario in fact falls in the framework with partially accessible servers considered by Foss and Chernova [7]. While the corresponding stability condition derived in [7] has a somewhat different form, it can be shown that it is equivalent to ours using the max-flow min-cut theorem of Ford–Fulkerson [6]. This proof is discussed in Sect. 5.2.

Remark 2 (*General applicability of the R-coupling*) The scope of application of the R-coupling is much broader than the general affinity-scheduling model. The described method is powerful enough to handle any assignment policy for the type-II jobs as long as the type-I jobs are assigned to a server with the shortest queue length within the primary selection. This implies that the number of type-I jobs will still be dominated by the number of jobs in the reference system. Furthermore, if the assignment policy for the type-II jobs only allows a finite number of them in each queue, then stability is guaranteed once $\lambda_0 < \mu_1$. For instance, this method could also be applicable in a setting without type-II jobs as the general compatibility model. In this setting, arriving jobs can only be served at predetermined server selections.

3.3 Neighborhood model

We will further investigate our model on a graph topology G_N as described in Sect. 2. It is challenging to get a grip on the performance of an assignment policy that is applied in a network structure, and establishing stochastic dominance relations can give an initial insight into the theoretical behavior of load balancing algorithms in structured environments. It is mentioned in Sect. 2 that the arrival rate over all server selections established by the graph structure G_N is given by λ, and, thus, Theorem 1 is still valid if we set $\lambda_0 = \lambda$. However, we will make two different assumptions on the structure of the graph topology, and for each of them a tighter dominance result than Theorem 1 is obtained. The first scenario assumes that the minimum degree of G_N is sufficiently high and the second scenario entails regular graph topologies.

3.3.1 Minimum degree

The reference system with N exchangeable servers operates under a modified version of the JSQ assignment policy, namely MJSQ(k) [19]. In this setting, new jobs arrive at a total rate of λN and are processed at a server according to a FCFS policy at rate $\mu_1 > \lambda$. An arriving job is assigned to the server with the $(k+1)$th shortest queue length. A clear analogy can be seen if the system is initially completely empty; then,

k servers will constantly remain idle. The system operates as if only $N-k$ servers are present and applies a JSQ policy restricted to these servers. If N is sufficiently large compared to k, i.e., if

$$\lambda N < \mu_1(N-k), \tag{7}$$

then the MJSQ(k) policy is stochastically stable.

Suppose that the minimum degree of the graph G_N is at least $N-k-1$, without any other structural assumptions. Letting N and k satisfy the relation in (7), we can describe a coupling between our neighborhood model with underlying topology G_N and the reference system with the MJSQ(k) policy. The coupling between both systems will fit the general framework of the affinity coupling. The coupling method for the arriving jobs will differ from the general setting in the previous subsection, and the coupling between the service completions follows the methodology explained in Sect. 3.1.

Coupling at arrival epochs For each of the neighborhood sets in $\mathcal{S}$, there is a uniform arrival rate λ such that the total arrival rate in the affinity system is also given by λN. Assuming that an event in the coupled sample path is an arrival, it is always directed to the server at position $k+1$ under the MJSQ(k) policy. For the neighborhood model, the primary selection S consists of a randomly selected server and its neighbors under the topology G_N and the secondary selection S^c contains all other servers. We do not know the exact ordered positions of the servers in the primary selection that is of size at least $N-k$ in terms of the $\overline{Q}_i$ variables. The worst-case scenario that could arise is a primary selection of size exactly $N-k$ where the servers are the $N-k$ highest ordered servers. Then, a type-I job is assigned to the server at position $k+1$. All other scenarios where an arriving job is labeled as a type-I job in the affinity system will lead to an assignment that is at most at the $(k+1)$th position. Hence, property (a) of the affinity coupling is satisfied.

Theorem 2 follows from the majorization result in Lemma 1 under the above-described coupling.

Theorem 2 (Neighborhood model with minimum degree $N-k-1$) *Consider the neighborhood model with an underlying graph topology with minimum degree $N-k-1$ and a reference system that operates under the* MJSQ(k) *policy. Then, for suitable initial conditions,*

$$\{(\overline{Q}^{\text{aff}}_{m+}(t))_{m\geq 1}\}_{t\geq 0} \leq_{\text{st}} \{(\overline{Q}^{\text{MJSQ}(k)}_{m+}(t))_{m\geq 1}\}_{t\geq 0}. \tag{8}$$

Once the reference system is stochastically stable, if condition (7) is fulfilled, we can give a meaningful upper bound on the total number of type-I jobs in the neighborhood model in terms of the total number of jobs under the MJSQ(k) policy. Note that the stability region in (7) for fixed values of N is smaller compared to the general setting in Sect. 3.2. However, when N is sufficiently large, the MJSQ(k) will outperform the RA policy, implying better performance results on the fluid level. So this upper bound will then be tighter compared to the result in Theorem 1.

Remark 3 Theorem 2 can be generalized for scenarios of the affinity-scheduling model where each server selection S has a size of at least $N-k$ and a non-uniform arrival rate λ_S.

3.3.2 Regular graph

As mentioned in introduction, JSQ(k) gives already substantial performance improvements for small values of k compared to the RA policy. With this in mind, we show that the number of type-I jobs under our affinity-scheduling policy on a d-regular graph is stochastically dominated by the total number of jobs under a JSQ(k) policy, when d and k satisfy the following relation:

$$\sum_{i=1}^{N-d-1} \binom{N-i}{k-1} \leq \frac{d+1}{N} \binom{N}{k}. \tag{9}$$

This condition manifests itself in the coupling construction between the arrival events in both systems such that feature (a) of the R-coupling is maintained. We will introduce a novel approach to represent or visualize all possible server selections in $\mathcal{S}$ that an arriving job can choose from.

An arriving job in the reference system with JSQ(k) is assigned to the lowest positioned server among k randomly selected servers. In total, there are $\binom{N}{k}$ server selections and each server belongs to $\binom{N-1}{k-1}$ different server selections. Thus, the lowest positioned server of the system belongs to $\binom{N-1}{k-1}$ different server selections; the second lowest server is part of precisely $\binom{N-2}{k-1}$ different server selections without the lowest ordered server. One can continue this reasoning up to the $(N-k+1)$th lowest ordered server; this server belongs to only one more server selection that is not yet observed at any of the lower-ordered servers. All higher-ordered servers cannot be part of an unobserved server selection. We will construct a step function from the positions $\{1, \ldots, N\}$ to the interval $[0, 1]$ based on the so-called *block interpretation* of the server selections. Assume that the servers are ordered from 1 to N and the lowest ordered position of the k selected servers is denoted by n. We represent this selection as a block from position n to N with height $1/\binom{N}{k}$. This procedure can be repeated for each of the $\binom{N}{k}$ possible selections, and these corresponding blocks can be stacked according to their length. This procedure is called the block interpretation of the server selections. The outer edges of these stacked blocks will give rise to the following step function:

$$f_{\text{ref}} \colon \{1, \ldots, N\} \to [0, 1] \colon x \mapsto \begin{cases} \frac{1}{\binom{N}{k}} \sum_{i=1}^{x} \binom{N-i}{k-1}, & 1 \leq x \leq N-k+1, \\ 1, & N-k+1 < x \leq N. \end{cases} \tag{10}$$

A visualization is shown in Fig. 2.

We aim to construct a similar step function based on the possible primary server selections for the affinity system when the underlying graph topology is a d-regular graph. Jobs arrive at a total rate λN, and an arriving job selects uniformly at random a server selection S from $\mathcal{S}$. By construction, $\mathcal{S}$ contains N different primary server selections, each of size $d+1$. Then, the lowest ordered server in the system belongs to $d+1$ different server selections. However, it is not possible to count the number of additional server selections containing the second lowest ordered server without

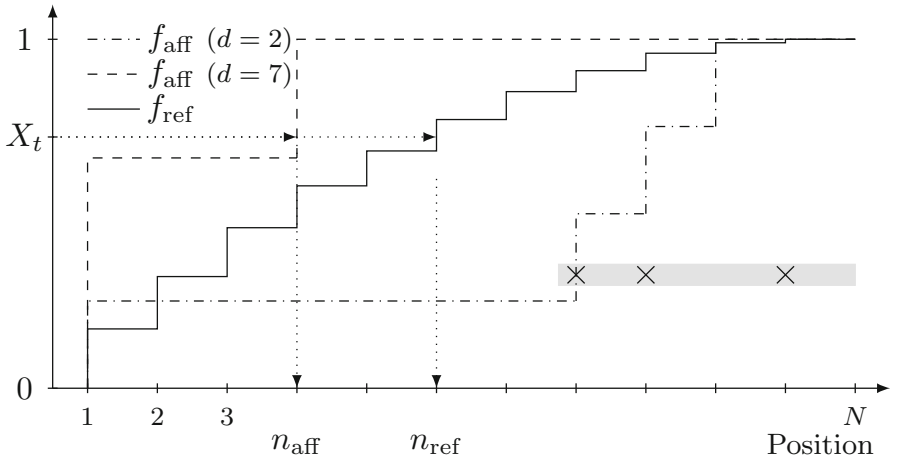


Fig. 2 A visualization of the step functions of the server selections in both systems. There are $N = 12$ servers, $d = 2$ (or $d = 7$) and $k = 3$. The block interpretation of server selection $S = \{8, 9, 11\}$ of the affinity system is represented with a gray block

knowing the position of each of the servers, since we operate on a fixed graph structure. We construct a step function based on the worst-case scenario to maintain property (a) of the coupling where the lowest positioned server of each of the server selections is still at the highest possible position. The first jump occurs at the lowest positioned server, while all remaining jumps will occur at the highest possible positioned servers. This induces stronger correlations between the servers that have more type-I jobs. Notice that the worst-case ordering of the servers might not align with the real underlying d-regular structure, so that this approach might be too conservative. The step function of this worst-case scenario is given by

$$f_{\text{aff}} \colon \{1, \dots, N\} \to [0, 1]$$
$$x \mapsto \begin{cases} \frac{d+1}{N}, & 1 \le x \le N - d - \lceil \frac{N}{d+1} \rceil + 1, \\ 1 - \frac{d+1}{N}(N - d - x), & N - d - \lceil \frac{N}{d+1} \rceil + 2 \le x \le N - d - 1, \\ 1, & N - d \le x \le N. \end{cases} \tag{11}$$

An example of this step function is shown in Fig. 2.

Coupling at arrival epochs Let the total arrival rate be λN in both systems. For an arriving job at time t, we determine the servers of interest using the inverse transform sampling method [4, Chapter 2]. First, we note that the functions f_{aff} and f_{ref} are cumulative distribution functions by construction. Second, the only server of interest of the server selection S in the affinity system or the server selection in the reference system is the lowest positioned server. So we sample a random variable X_t from a uniform distribution on $[0, 1]$ and determine the two servers positions, n_{aff} and n_{ref}, of interest of both systems. This procedure is visualized in Fig. 2. In the affinity system,

Table 1 Smallest possible value of d, d^*, that satisfies condition (9) is listed for a system with $N = 50$ servers and given values of k

k	2	3	4	5	10	15	25
d^*	31	34	36	38	42	44	46

a job can be assigned as a type-I job to the selected server, or it may be assigned to any other server as a type-II job.

So in order to guarantee feature (a) of the affinity coupling, it needs to be ensured that $n_{\text{aff}} \leq n_{\text{ref}}$, i.e., $f_{\text{aff}}(n) \geq f_{\text{ref}}(n)$ for all positions n. We observe that the d-regular graph must be rather dense in order to obtain a tighter upper bound than provided by the RA policy. Once the degree d is at least $N/2$, the step function f_{aff} only makes two jumps, at positions 1 and $N-d$ of sizes $(d+1)/N$ and $(N-d-1)/N$, respectively. Since the step function f_{ref} is concave in its discrete points, we only need to ensure that $f_{\text{aff}}(N-d-1) \geq f_{\text{ref}}(N-d-1)$ holds, so that the step function of the affinity system is above the step function of reference system. This results in condition (9) on the values of d and k. Due to the coupling construction, we can prove the following dominance result.

Theorem 3 (Neighborhood model with $\boldsymbol{d}$-regular graph) *Consider the neighborhood model with an underlying d-regular graph topology and a reference system operating under a* JSQ(k) *policy. If the model parameters d and k satisfy condition* (9)*, then, for suitable initial conditions,*

$$\{(\overline{Q}_{m+}^{\text{aff}}(t))_{m\geq 1}\}_{t\geq 0} \leq_{\text{st}} \{(\overline{Q}_{m+}^{\text{JSQ}(k)}(t))_{m\geq 1}\}_{t\geq 0}. \tag{12}$$

Due to the coupling construction using the block interpretation of the server selections, the above-described coupling fits the general framework of the affinity coupling as stated in Lemma 1. Therefore, the result of Theorem 3 follows from this lemma. If $\lambda < \mu_1$, the reference system is stochastically stable and provides a meaningful upper bound on the performance of the neighborhood model on a d-regular topology.

In Table 1, we list d^*, the minimum value of d as a function of k that guarantees the required dominance of the step functions for a system with $N = 50$ servers. Furthermore, Fig. 3 visualizes the behavior of d^* for fixed values of k and increasing graph sizes N. The minimum degree seems to grow linearly with N. For instance, it is a straightforward computation to show that

$$d^* = \left\lceil \frac{\sqrt{N^2 + 4(N-1)^2} - N}{2} \right\rceil \tag{13}$$

when $k = 2$. This expression is indeed of linear order in N.

From Table 1 and Fig. 3, we observe that the graph structure must be rather dense in order to stochastically dominate the affinity-scheduling policy with a JSQ(k) policy even for small values of k. One can argue that the primary selection of our affinity-scheduling strategy must be much larger compared to the server selection under JSQ(k)

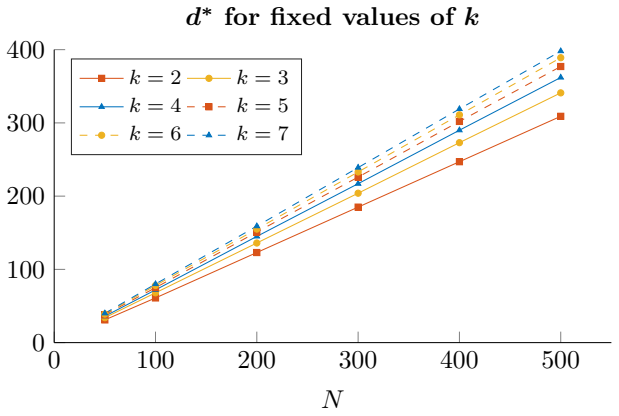


Fig. 3 A visualization of the smallest possible value of d, d^*, that satisfies condition (9) as function of N for fixed values of k

in order to guarantee better performance. But we should keep in mind that the underlying graph structure is fixed and all possible server selections are predetermined, while the JSQ(k) strategy can be seen as a strategy on a complete graph where an arbitrary set of size k of the servers can be selected. This will affect the performance compared to a system with N exchangeable servers, which is intuitively clear.

Moreover, it is important to note that the obtained value of d^* might be too conservative. Our coupling method using the step functions requires a degree that is at least equal to $N/2$ in order to upper bound by the strategy JSQ(1), i.e., the RA policy. On the other hand, we showed in the general result that the number of type-I jobs under any structural interpretation $\mathcal{S}$ is stochastically dominated by the number of jobs under a random assignment strategy.

Remark 4 (*Combinatorial model*) Applying our affinity-scheduling strategy to the combinatorial model with N servers shows a lot of similarities with the JSQ(d) policy in the setting of N exchangeable servers. Namely, an arriving job is assigned to the server with the shortest queue length among d arbitrarily selected servers and sometimes the job can be directed to an idle server outside this selection. The coupling can be adjusted such that the number of type-I jobs at the nth ordered position under our affinity-scheduling policy is less than or equal to the number of jobs at the nth ordered position under a JSQ(d) policy. The relation between the two policies will become more apparent in Sect. 4 when fluid limit results are investigated.

Moreover, it can be shown that the combinatorial model is stochastically stable under a preemptive and a non-preemptive scheduling policy using a Foster–Lyapunov argument. This result is shown under the assumption that the number of type-II jobs at each server never exceeds one. The fact that stability is preserved under a preemptive policy in favor of the type-I jobs is no surprise due to the structure of the server selections $\mathcal{S}$ and the resemblance of the first step in the assignment policy with the JSQ(d) policy. Under a non-preemptive policy, it is no longer intuitively clear, as any finite value of μ_2 is allowed and one could imagine a situation where all servers are processing a type-II job and type-I jobs start to accumulate behind these type-II jobs.

4 Fluid limit and fixed point analysis

As mentioned in the introduction, the affinity model in general lacks the exchangeability among the servers that underpins the use of mean field limits as the main analytical techniques in the supermarket model. Due to its inherent symmetry, the combinatorial model with uniform arrival rates for each of the server selections in $\mathcal{S}$ as described in Sect. 2 is one of the exceptions. The variables $(Q_{ij}^N(t))_{i,j}$ will give rise to a Markov process representation in this case. The primary and secondary server selections for an arriving job are of sizes d and $N-d$, respectively. In order to gain insight into the system performance, we introduce the fluid-scaled variables, i.e.,

$$\left(\frac{\overline{Q}_{ij}^N(t)}{N}\right)_{i,j},$$

and analyze a sequence of systems where the number of servers N tends to infinity. The (weak) limit that arises is referred to as the fluid limit and is denoted by $(\overline{q}_{ij}(t))_{i,j}$. When it is helpful to stress the proportion of servers with exactly i type-I jobs, instead of at least i type-I job, we consider the variables $(q_{ij}(t))_{i,j}$. It is clear that $q_{ij}(t)$ is given by $\overline{q}_{ij}(t)-\overline{q}_{i+1,j}(t)$ for any i and j. Furthermore, we assume that $\lambda < \mu_1$ to guarantee stochastic stability. Throughout this section, we will consider a system with $\lambda = 0.8$, $\mu_1 = 1$ and $\mu_2 = 0.5$ in the numerical and simulation experiments, unless specified otherwise.

4.1 Fluid limit

We now provide a characterization of the (deterministic) fluid limit in terms of a set of discontinuous differential equations. The t reference in the notation will be omitted, if the context allows this.

We introduce a reduced arrival rate $\tilde{\lambda}$. A job will always be directed to an idle server if available, either as a type-I job or a type-II job, and idle servers are generated at rate $\mu_1 q_{10} + \mu_2 q_{01}$. This implies that if λ is sufficiently high, i.e., $\lambda > \mu_1 q_{10} + \mu_2 q_{01}$, only a fraction of the arriving jobs will start to queue in front of a server as type-I jobs on the fluid level. This fraction is given by $\tilde{\lambda}/\lambda$, with

$$\tilde{\lambda} = (\lambda - \mu_1 q_{10} - \mu_2 q_{01})^+. \tag{14}$$

Then,

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{00} = \mu_2 q_{01} - \lambda(1-q_{00})^d + \tilde{\lambda}\mathbb{1}\{q_{00}=0\}, \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{01} = -\mu_2 q_{01} + \mathbb{1}\{q_{00}>0\}\left[\lambda(1-q_{00})^d\right] + \mathbb{1}\{q_{00}=0\}\left[\lambda-\tilde{\lambda}\right], \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{10} = -\mu_1 q_{10} + \mathbb{1}\{q_{00}>0\}\left[\lambda\left(1-(1-q_{00})^d\right)\right], \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{11} = -\mu_1 q_{11} + \tilde{\lambda}\mathbb{1}\{q_{00}=0\}\left[\left(\overline{q}_{10}+\overline{q}_{01}\right)^d - \left(\overline{q}_{10}+\overline{q}_{11}\right)^d\right], \\ \text{for } i \geq 2, \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{i0} = -\mu_1 q_{i0} + \tilde{\lambda}\mathbb{1}\{q_{00}=0\}\left[\left(\overline{q}_{i-1,0}+\overline{q}_{i-1,1}\right)^d - \left(\overline{q}_{i0}+\overline{q}_{i-1,1}\right)^d\right], \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{i1} = -\mu_1 \overline{q}_{i1} + \tilde{\lambda}\mathbb{1}\{q_{00}=0\}\left[\left(\overline{q}_{i0}+\overline{q}_{i-1,1}\right)^d - \left(\overline{q}_{i0}+\overline{q}_{i1}\right)^d\right], \end{cases} \tag{15}$$

with $\overline{q}_{00} + \overline{q}_{01} = 1$.

Since the system operates under a preemptive priority policy, the structure of the departure rate in each of the equations in (15) is clear. For instance, in order to change the proportion $\overline{q}_{11}$ due to a job completion, this job completion must take place at a server with configuration $(1, 1)$. Exactly a fraction q_{11} of the servers has this configuration, and since these servers each work at rate μ_1, the total rate of change is given by $-\mu_1 q_{11}$.

Let us illustrate the representation of the arrival term for the derivative of $\overline{q}_{11}$. Only an arrival of a type-I job at a server with configuration $(0, 1)$ can contribute to the arrival term, and the probability that this configuration is the smallest among the d servers in the primary selection is given by $\left(\overline{q}_{10}+\overline{q}_{01}\right)^d - \left(\overline{q}_{10}+\overline{q}_{11}\right)^d$. Ties are broken according to the presence of a type-II job, in favor of having no type-II jobs. Moreover, there should be no idle servers because otherwise an arriving job would be assigned here as a type-II job. Since type-I jobs arrive at a reduced rate $\tilde{\lambda}$, the total rate of change is given by $\tilde{\lambda}\mathbb{1}\{q_{00}=0\}[\left(\overline{q}_{10}+\overline{q}_{01}\right)^d - \left(\overline{q}_{10}+\overline{q}_{11}\right)^d]$.

The expressions for the arrival terms in the derivatives of (15) and the reduced arrival rate $\tilde{\lambda}$ should be considered more carefully due to the discontinuity at $q_{00} = 0$. We will give a sketch of the derivation of this fluid limit in Sect. 5.3. This derivation relies on the martingale method for point processes and Markovian queueing settings outlined by Pang et al. [21] and Brémaud [3].

The fluid limit expression can be validated with simulations of the fluid-scaled stochastic process. Consider, for instance, Fig. 4, where the solution of the fluid limit (15) is presented together with a simulated trajectory of a reasonably large system. It can be observed that the simulated trajectory fluctuates closely around the numerical solution of the fluid limit, which supports the connection between the fluid limit and the behavior of the stochastic system in a many-server setting.

4.2 Fixed points

To investigate the long-run behavior of the fluid limit (15), we are interested in its fixed points. It turns out that the mutual relationships between the model parameters d, λ, μ_1 and μ_2 play a crucial role. In the remainder of this section, we investigate the

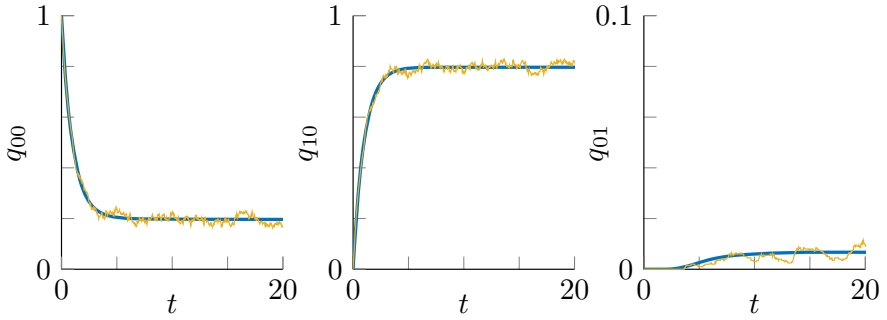


Fig. 4 A comparison between a simulated trajectory of the fluid-scaled stochastic process with $N = 2000$ servers (thin line) and the numerical solution (thick line) of the fluid limit (15). The model parameters are given by $d = 25$, $\lambda = 0.8$, $\mu_1 = 1$ and $\mu_2 = 0.5$

setting where $\lambda > \mu_2$, in order to compare one of the fixed points with the fixed point of a JSQ(d) policy with reduced load $\tilde{\lambda}$.

Theorem 4 (Fixed points) *When $\lambda > \mu_2$ and $d \geq 2$, the system of differential equations (15) always has the following fixed point:*

$$\begin{cases} \overline{q}^*_{i0} = 0, \\ \overline{q}^*_{i1} = \left(\frac{\lambda-\mu_2}{\mu_1-\mu_2}\right)^{\frac{d^i-1}{d-1}}, \quad i = 0, 1, 2 \ldots. \end{cases} \tag{16}$$

Let $d^* \doteq d^*(\lambda, \mu_1, \mu_2)$ denote the minimum selection size that satisfies

$$\begin{aligned} & d^*\lambda\left(\frac{1}{\mu_2} - \frac{1}{\mu_1}\right) > 1, \\ & \left(1 - \frac{1}{d^*}\right)\frac{\mu_1}{\lambda} > \left(d^*\lambda\left(\frac{1}{\mu_2} - \frac{1}{\mu_1}\right)\right)^{\frac{1}{d^*-1}}. \end{aligned} \tag{17}$$

If $d \geq d^*(\lambda, \mu_1, \mu_2)$, then precisely two more fixed points exist. These fixed points are such that $q_{00} + q_{01} + q_{10} = 1$ and $q_{00} > 0$.

The proof of this theorem is discussed in Sect. 5.3. It can be observed that there always exists a sufficiently large d value that satisfies both inequalities of condition (17) for given values of λ, μ_1 and μ_2. This is trivial to see for the first inequality. The second inequality can be rewritten as

$$\left(1 - \frac{1}{d}\right)\left(\frac{1}{d \cdot a}\right)^{\frac{1}{d-1}} > \frac{\lambda}{\mu_1} \tag{18}$$

with $a \doteq \lambda(\frac{1}{\mu_2} - \frac{1}{\mu_1})$. The left-hand side is increasing as a function of d with limit $1 > \lambda/\mu_1$. Table 2 gives the values of d^* satisfying (17) for a range of model parameters. It can be seen that the higher the load, the larger the size of the primary selections

Table 2 For given λ, $\mu_1 = 1$ and μ_2, the minimum value d^* that satisfies condition (17) is listed

λ	0.4	0.5	0.6	0.7	0.8	0.9
$\mu_2 = 1/2$	/	/	5	9	18	46
$\mu_2 = 1/3$	3	5	7	12	22	54

must be for multiple fixed points to persist. The additional fixed points have a strictly positive fraction of idle servers, and it is intuitively clear that the number of servers where a job can be processed at rate μ_1 must grow with the load in order for such fixed points to persist.

The long-term fraction of servers with at least i jobs under a JSQ(d) policy is given by

$$\overline{q}_i^* = \left(\frac{\tilde{\lambda}}{\mu_1}\right)^{\frac{d^i-1}{d-1}} = \left(\frac{\lambda - \mu_2}{\mu_1 - \mu_2}\right)^{\frac{d^i-1}{d-1}}, \tag{19}$$

for $i \geq 0$, with arrival rate $\tilde{\lambda}$ and service rate μ_1 [18]. This shows a strong similarity with the fixed point (16) where two types of jobs are taken into account. Next we consider the case $d = 1$. When $\lambda > \mu_2$, there still is a unique fixed point with $q_{00} = 0$, given by

$$\begin{cases} \overline{q}_{i0}^* = 0, \\ \overline{q}_{i1}^* = \left(\frac{\lambda-\mu_2}{\mu_1-\mu_2}\right)^i, \quad i = 0, 1, 2, \ldots. \end{cases} \tag{20}$$

This shows strong resemblance with the RA policy with load $\rho = \tilde{\lambda}/\mu_1$. Allowing a primary selection of at least two servers leads to a super-exponential improvement compared to a primary selection of size one. On the other hand, there is no fixed point with $\lambda \geq \mu_2$ and $q_{00} > 0$. Only if $\lambda < \mu_2$ can we show that there is a unique fixed point with $q_{00} > 0$, namely

$$q_{00}^* = \frac{(\mu_2 - \lambda)\mu_1}{(\mu_2 - \lambda)\mu_1 + \lambda\mu_2}. \tag{21}$$

4.3 Further analysis

We will conduct a further analysis of the fluid limit (15) where we distinguish between $d < d^*$ and $d \geq d^*$.

4.3.1 Sufficiently small primary selections

When d is sufficiently small in terms of the model parameters λ, μ_1 and μ_2, i.e., $d < d^*$, the fixed point (16) of the fluid limit (15) is unique. Numerical experiments suggest that this fixed point is a global attractor, i.e., the trajectories of the fluid limit will converge to this fixed point for every initial state of the system. As an example, we present Fig. 5, where the numerical solution of the fluid limit is visualized for

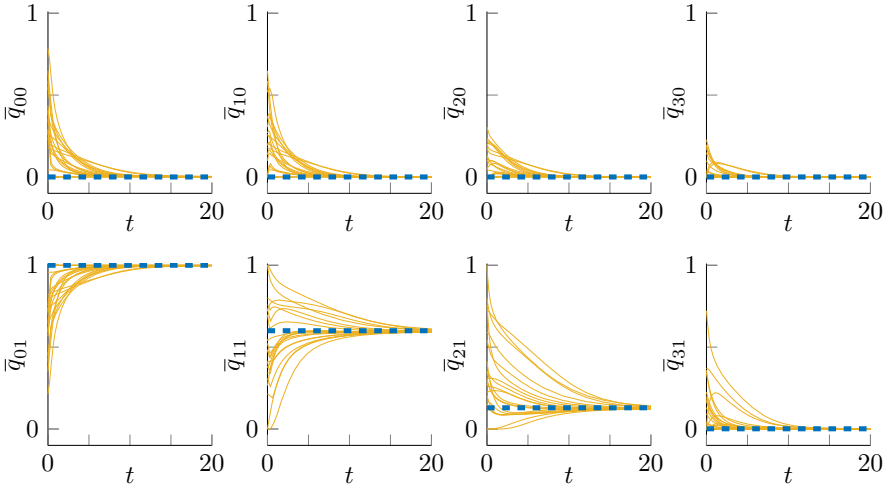


Fig. 5 Ten solution trajectories (thin lines) of the fluid limit (15) are plotted for randomly generated initial states, showing convergence of the cumulative distributions to the fixed point (thick dashed line). The model parameters are given by $d = 3$, $\lambda = 0.8$, $\mu_1 = 1$ and $\mu_2 = 0.5$, and each of the servers has at most three type-I jobs in the initial states

ten randomly sampled initial configurations. We consider a system with the above-mentioned model parameters and a primary selection of size $d = 3$. As can be seen from the figure, all cumulative fractions $(\overline{q}_{i0})_{i\geq 0}$ tend to zero. The fractions $\overline{q}_{01}$, $\overline{q}_{11}$, $\overline{q}_{21}$ and $\overline{q}_{31}$ converge to 1, 0.6, 0.1296 and 0.0013, respectively.

In the previous section, we used the R-coupling to show stochastic stability and the existence of an (unknown) stationary distribution for $\lambda < \mu_1$. Assuming global stability of the unique fixed point, Theorem 1 by Benaïm and Le Boudec [2] ensures that the large-N limit of the stationary distribution will converge to the fixed point. Moreover, from simulations it can be observed that the trajectories indeed converge to the unique fixed point of the fluid limit (15). As an example, consider a system with the above-mentioned model parameters. Figure 6 shows a simulated trajectory of the fluid-scaled variables for a system with $N = 2000$ servers that is initially completely empty. It can be seen that the trajectory converges to the fixed point $(q_{01}, q_{11}, q_{21}, \dots) = (0.40, 0.4704, 0.1283, \dots)$, rounded at four decimals.

The asymptotic approximation for the mean stationary queue length, excluding the job in service, suggested by the fixed point is given by

$$\mathbb{E}\left[Q_{\mathrm{CM}(d)}\right] = \sum_{i\geq 1} i q_{i,1} = \sum_{i\geq 1} \left(\frac{\lambda - \mu_2}{\mu_1 - \mu_2}\right)^{\frac{d^i-1}{d-1}}. \tag{22}$$

Here CM(d) refers to the combinatorial model with a primary server selection of size d. It might be interesting to compare the performance of our affinity-scheduling policy with the performance of the JSQ(d) policy in an ordinary supermarket model with arrival rate λ and service rate μ_1 [18,27]. The combinatorial model can be seen as an

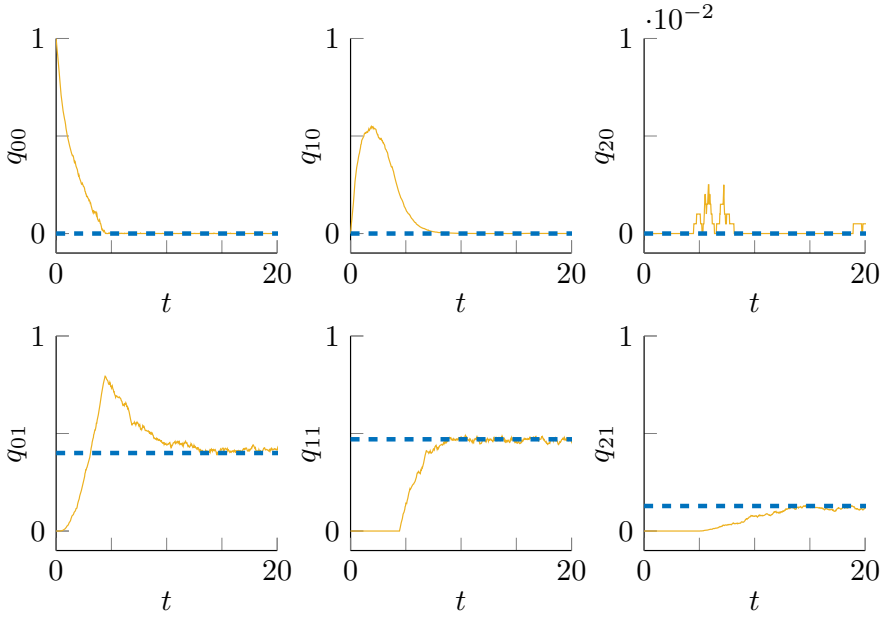


Fig. 6 A comparison between the long-term behavior of a simulated trajectory for $N = 2000$ servers (thin line) and the unique fixed point (thick dashed line) of the fluid limit (15). The model parameters are given by $d = 3$, $\lambda = 0.8$, $\mu_1 = 1$ and $\mu_2 = 0.5$

extension of the JSQ(d) policy with the additional feature that jobs can be assigned to a distant server if this allows the service to start immediately. The mean queue length under the JSQ(d) policy is given by

$$\mathbb{E}\left[Q_{\mathrm{JSQ}(d)}\right] = \sum_{i\geq 1} i q_{i+1} = \sum_{i\geq 1} \left(\frac{\lambda}{\mu_1}\right)^{\frac{d^{i+1}-1}{d-1}}. \tag{23}$$

Furthermore, the exact mean queue length under the RA policy is given by

$$\mathbb{E}\left[Q_{\mathrm{RA}}\right] = \left(1 - \frac{\lambda}{\mu_1}\right) \sum_{i\geq 1} i \left(\frac{\lambda}{\mu_1}\right)^{i+1} = \frac{(\lambda/\mu_1)^2}{1 - \lambda/\mu_1}. \tag{24}$$

Figure 7 presents a comparison of the number of waiting jobs as a function of λ, with $d = 3$, $\mu_1 = 1$ and $\mu_2 = 0.5$. It is known that the mean queue length for the RA policy tends to infinity when the offered traffic grows to one. We see that the mean queue length in the combinatorial model is slightly larger than for the JSQ(d) policy. On the other hand, the variance of the queue length in the JSQ(d) model is almost twice as large compared to the combinatorial model. We conclude that the combinatorial model still performs well from a queue length perspective, even though each server has a type-II job and possibly multiple type-I jobs in its queue.

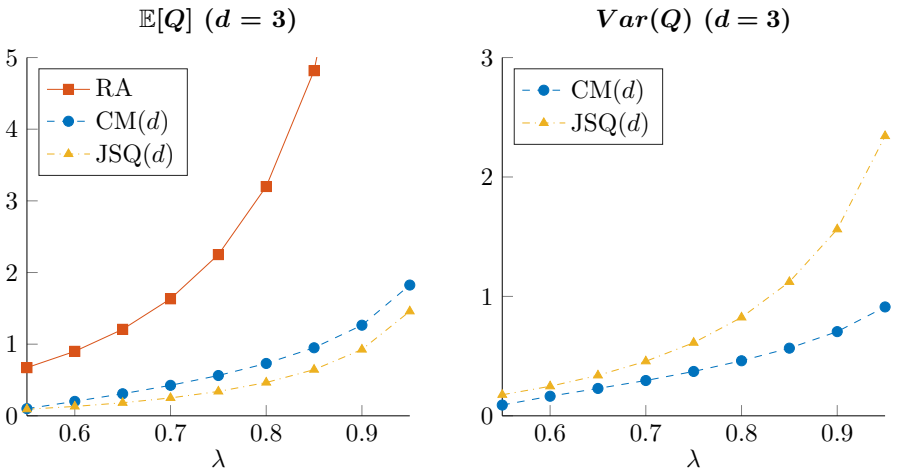


Fig. 7 Comparison of the mean and variance of the queue length as a function of λ between three different models on the fluid level for $d = 3$, $\mu_1 = 1$ and $\mu_2 = 0.5$

From the fixed point expression, it is not immediately visible that type-II jobs finish their service, since the fraction q_{00} is zero. However, an idle server will be filled instantly with an arriving job. A total fraction

$$\frac{\lambda - \tilde{\lambda}}{\lambda} = \frac{\mu_2}{\lambda} \frac{\mu_1 - \lambda}{\mu_1 - \mu_2} \tag{25}$$

of the arriving jobs undergo this 'immediate switch': they are assigned as a type-II job to a server that just emptied its queue. This fraction decreases in λ, and for example, for the above-mentioned model parameters, this leads to a fraction of $1/4$. Furthermore, the type-II jobs will leave the system at the same rate $\lambda - \tilde{\lambda}$ as they enter the system, since we study the system in equilibrium. Moreover, due to Little's law we know that the expected waiting time of an arbitrary job is finite. Let W denote the waiting time, then $\mathbb{E}\left[Q_{\mathrm{CM}(d)}\right] = \lambda \mathbb{E}[W]$. Since the expected queue length under our affinity-scheduling policy is finite, this results in a finite expected waiting time for an arbitrary job, so also for the type-II jobs.

Since each server operates under a preemptive scheduling policy, we can calculate the average waiting time of a type-I job using Little's law. Let Q_{I} denote the number of type-I jobs at a server. Then,

$$\mathbb{E}[Q_{\mathrm{I}}] = \sum_{i \geq 1} i q_{i+1,1} = \sum_{i \geq 1} \left(\frac{\lambda - \mu_2}{\mu_1 - \mu_2} \right)^{\frac{d^{i+1}-1}{d-1}}. \tag{26}$$

Furthermore, the reduced arrival rate $\tilde{\lambda}$ gives the arrival rate of type-I jobs on the fluid level. If W_{I} represents the waiting time of a type-I job, then due to Little's law $\mathbb{E}[Q_{\mathrm{I}}] = \tilde{\lambda} \mathbb{E}[W_{\mathrm{I}}]$. Let Q_{II} and W_{II} have the same interpretation as above but for the

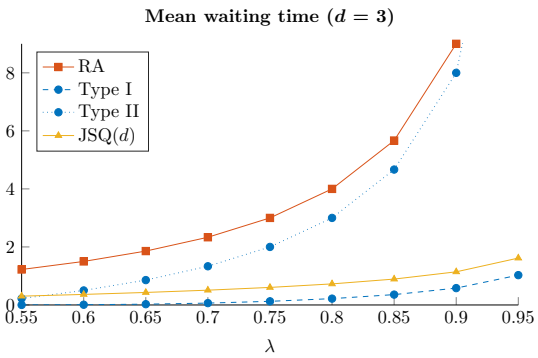


Fig. 8 A comparison of the mean waiting times of the type-I and type-II jobs with the mean waiting times under the RA policy or a JSQ(d) policy as function of λ for $d = 3$, $\mu_1 = 1$ and $\mu_2 = 0.5$

type-II jobs. We condition on the type of job to obtain

$$\mathbb{E}[W] = \frac{\tilde{\lambda}}{\lambda}\mathbb{E}[W_{\mathrm{I}}] + \frac{\lambda - \tilde{\lambda}}{\lambda}\mathbb{E}[W_{\mathrm{II}}]. \tag{27}$$

Because of Little's law, this results in

$$\mathbb{E}[W_{\mathrm{II}}] = \frac{1}{\lambda - \tilde{\lambda}}\left(\mathbb{E}\left[Q_{\mathrm{CM}(d)}\right] - \mathbb{E}[Q_{\mathrm{I}}]\right) = \frac{1}{\lambda - \tilde{\lambda}}\frac{\lambda - \mu_2}{\mu_1 - \mu_2}. \tag{28}$$

We can also immediately apply Little's law to the type-II jobs. We know that they arrive at rate $\lambda - \tilde{\lambda}$ and the mean waiting queue length is by definition given by

$$\mathbb{E}[Q_{\mathrm{II}}] = \sum_{i \geq 1} q_{i1} = 1 - q_{01} = \frac{\lambda - \mu_2}{\mu_1 - \mu_2}. \tag{29}$$

In Fig. 8, we compare the mean waiting time of a type-I or type-II job with the mean waiting time under the RA or the JSQ(d) policy. The mean waiting time of type-II jobs is fairly high, but still lower than the waiting time under the RA policy. We also observe that the mean waiting time of type-I jobs is significantly smaller than under a JSQ(d) policy. We conclude that our assignment policy leads to a reduction in the mean waiting time for a large group of arriving jobs at the expense of some other jobs that encounter longer waiting times. The uniqueness of the fixed point allows us to analyze the asymptotic stationary distribution of the model. On the other hand, we observe that the value of the size of the server selection d is too small to achieve a zero waiting time for an arriving job.

4.3.2 Sufficiently large primary selections

Assume that the primary selection has a sufficiently large size d for given model parameters in terms of the conditions (17), i.e., $d \geq d^*$. From Theorem 4, we know that, in addition to the closed-form fixed point (16), there are two more fixed points with $q_{00} + q_{01} + q_{10} = 1$. We prove the following theorem using the indirect Lyapunov method.

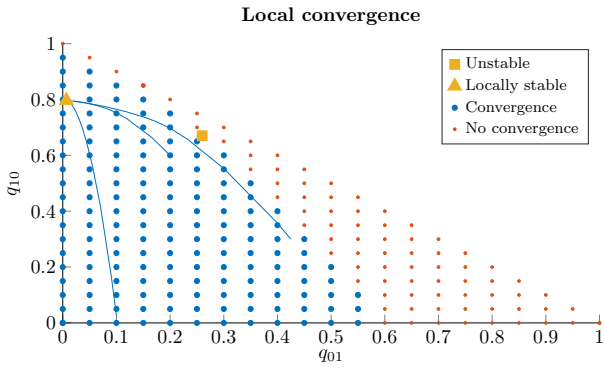


Fig. 9 An overview of the initial states with $q_{00}+q_{01}+q_{10}=1$, if the corresponding trajectories converge to the locally stable fixed point (triangle), then the initial states are indicated with a large dot, and otherwise, they are indicated with a small dot. We consider a system with $d=25$, $\lambda=0.8$, $\mu_1=1$ and $\mu_2=0.5$

Theorem 5 (Local (in)stability) *Of the two additional fixed points mentioned in Theorem 4 with $q_{00}+q_{01}+q_{10}=1$ when $d \geq d^*$, one is locally stable and the other one is unstable.*

The proof of Theorem 5 is given in Sect. 5.3. In the remainder of this subsection, we will provide a numerical illustration, where we consider a system with $\lambda=0.8$, $\mu_1=1$, $\mu_2=0.5$ and $d=25$ throughout. We observed similar qualitative behavior across many different scenarios, but only present results for the above parameter values because of space constraints. To get a better notion of the local stability, we present Fig. 9. For several initial values such that $q_{00}+q_{01}+q_{10}=1$, the system of differential equations (15) is solved numerically. All trajectories with initial states indicated with large dots will converge to the locally stable fixed point from the previous theorem and a few of these trajectories are also visualized. All other initial states, indicated with small dots, will not converge to this locally stable fixed point. We see that these states have a large fraction of servers with a type-II job present and a small fraction of idle servers, since there is a smaller probability of selecting an idle server in the primary selection. So jobs will have a longer mean service time as a type-II job and jobs will start to accumulate.

In total, this gives rise to two locally stable fixed points: the closed-form fixed point (16) where each server has a type-II job and possibly multiple type-I jobs, and the fixed point from Theorem 5 where at most one job is present at each server. In the remainder of this section, we will refer to these fixed points as the *queueing fixed point* and *no-queueing fixed point*, respectively. We do not formally prove this statement, but we will illustrate it with a representative example. For a system with the above-mentioned parameters, the two fixed points under consideration (non-cumulative fractions) are given by

$$\begin{aligned}(q_{00}, q_{01}, q_{10}) &= (0.1966, 0.0067, 0.7967),\\ (q_{01}, q_{11}) &= (0.4, 0.6).\end{aligned} \tag{30}$$

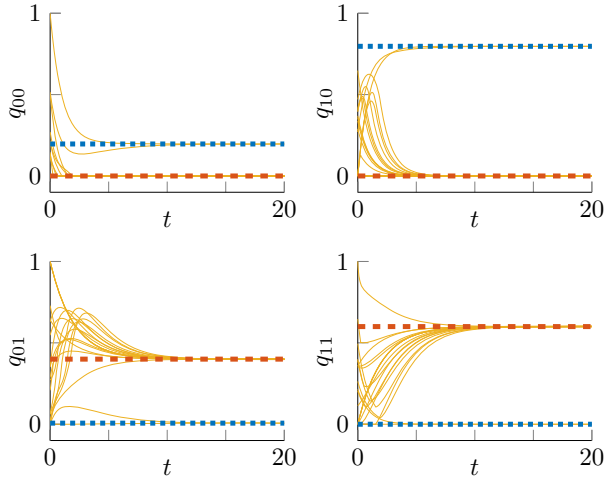


Fig. 10 Trajectories for 20 randomly sampled initial states are visualized, with initial points such that $q_{00} + q_{01} + q_{10} + q_{11} = 1$. The trajectories converge to one of the two fixed points; the no-queueing and queueing fixed point are indicated with a thick dotted line and thick dashed line, respectively

Both fixed points are indicated in Fig. 10 with dotted and dashed lines, respectively. Furthermore, the graphs contain 20 trajectories starting from randomly sampled initial configurations with $q_{00} + q_{01} + q_{10} + q_{11} = 1$; all these trajectories converge to one of the two fixed points. This implies that the convergence area presented in Fig. 9 to the no-queueing fixed point will in fact be larger. As can be seen, most of the trajectories will converge to the queueing fixed point. This phenomenon will be even more apparent if we allow initial states with more than two jobs.

The literature often describes systems with a unique global attractor as a fixed point of the fluid limit so that there is a direct connection between the stationary distribution in a many-server setting and this fixed point. However, the non-uniqueness of the fixed points does not imply that these two concepts are completely disjoint. For instance, Fig. 4 presents a comparison between the numerical solution of the fluid limit and a simulation with $N = 2000$ servers with the above-mentioned model parameters. The system is initially empty, and the simulated trajectory seems to converge to the no-queueing fixed point. We could present a similar figure, where in the initial configuration each server has one type-II job, in which case both the numerical solution and the simulation seem to tend to the queueing fixed point.

However, the stochastic process with a finite number of servers is an irreducible Markov process which implies that any state can be reached as long as the process is observed long enough and a unique equilibrium distribution must exist. Nevertheless, it can be observed that the residence time near each of the locally stable fixed points, which increases with N, is long before the process makes the transition to the other locally stable fixed point. Gibbens et al. [11] describe this concept of switching between multiple modes by 'tunneling.'

Examples of models where metastability plays an important role in loss and communication networks can be found in [1,11,30]. More recent work by Martirosyan and

Robert [17] considers an assignment policy closely related to the affinity-scheduling policy in a loss network setting, i.e., jobs can be redirected to distant servers with a penalty or can be omitted if none of the servers has enough spare capacity. Also in this setting, a fluid limit analysis reveals multiple locally stable fixed points.

The combinatorial model was introduced as a highly symmetric model instance of the more general affinity-scheduling model. This built-in symmetry allowed us to conduct a fluid limit analysis. However, simulation results indicate that the observed metastability is a more general phenomenon. The simulations were conducted for server systems with various sizes and various underlying graph structures, including 24-regular graphs and Erdös-Rényi random graphs with average degree 25. The remaining model parameters were kept as $\lambda = 0.8$, $\mu_1 = 1$ and $\mu_2 = 0.5$, and the system was initially completely empty. Note that for both instances of the neighborhood model the (average) size of the primary selection is at least 25 as before, but the variability among the possible primary selections has been greatly reduced compared to the combinatorial model instances. We chose for an average degree of 25 instead of 24 for the Erdös-Rényi random graphs to ensure that most of the sizes of the primary selections indeed satisfy condition (17). The value of the corresponding no-queueing fixed point when $d = 26$ slightly changes to $(q_{00}, q_{01}, q_{10}) = (0.1974, 0.0053, 0.7973)$. There is no noticeable difference for the value of the queueing fixed point up to four decimals. As an illustrative example, we present Fig. 11. To obtain this figure, 300 simulation runs were conducted for Erdös–Rényi random graphs consisting of $N = 250$ servers. The long-term fractions q_{00}, q_{10}, q_{01} and q_{11} were monitored and presented in a histogram. The values of the two above-mentioned locally stable fixed points are indicated with dotted and dashed lines, respectively. It can be observed that the long-term fractions in this non-symmetric setting still converge to either of the two fixed points. For increasing values of N, the simulated long-term fractions will be even closer to the theoretical fixed points of the combinatorial model.

5 Proofs

5.1 Proof of Lemma 1: R-coupling

Since the system configurations between two consecutive events remain unchanged, we will condition on the discrete event times and use forward induction.

Assume that (3) holds up to the time of the $(k-1)$th event. We will argue that the majorization property still holds at time t_k of the kth event by making a distinction between arrival and departure epochs. But first we need a formal way to express the effect of these events in terms of $(\overline{Q}_i^{\text{aff}}(t))_{i\geq 1}$ and $(\overline{Q}_i^{\text{ref}}(t))_{i\geq 1}$. For instance, the server with the nth shortest queue length is selected for a departure. Due to the ordering, we know that there are at least $N-n+1$ servers with the same number of jobs or more in their queues as the server at ordered position n. It might also be possible that the server at the ordered position $n-1$ has the same number of jobs as the server at position n. Then, there is no notable difference in terms of the variables $\overline{Q}_i$ whether a removal takes place at position $n-1$ or at position n. Instead of removing from the server at

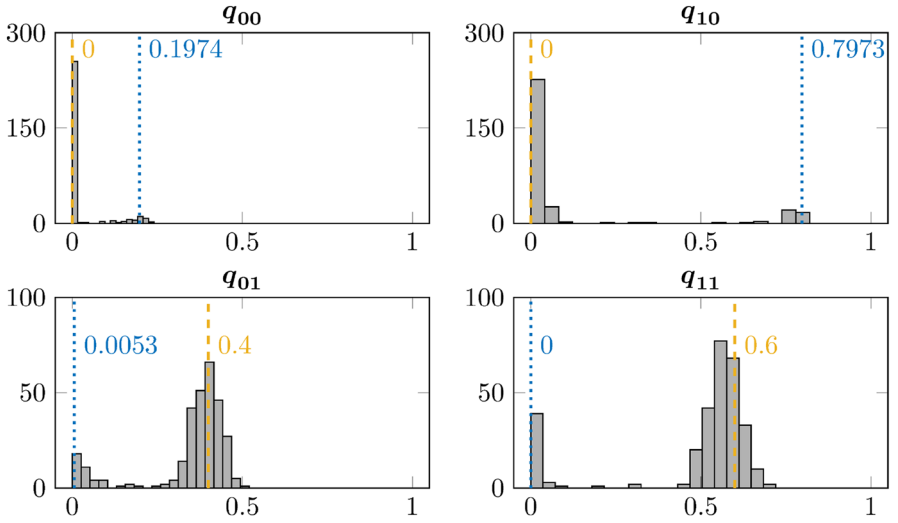


Fig. 11 Long-term fractions q_{00}, q_{10}, q_{01} and q_{11} of 300 simulation runs on Erdös–Rényi random graphs with average degree 25 and $N = 250$ servers. Indicated with dotted and dashed lines are the theoretical locally stable fixed points for combinatorial model instances with model parameters $d = 26$, $\lambda = 0.8$, $\mu_1 = 1$ and $\mu_2 = 0.5$

position n and reordering the servers before computing $(\overline{Q}_i^{\text{aff}}(t))_{i\geq 1}$ and $(\overline{Q}_i^{\text{ref}}(t))_{i\geq 1}$, we can also immediately update these variables. The difference is subtle and valid because the proof does not rely on the present type-II jobs or on the actual servers but only on their relative positions. Therefore, we define two intermediate quantities:

$$\begin{aligned} I_{\text{aff}}(n) &\doteq \max\{j : \overline{Q}_j^{\text{aff}} \geq N - n + 1\}, \\ I_{\text{ref}}(n) &\doteq \max\{j : \overline{Q}_j^{\text{ref}} \geq N - n + 1\}. \end{aligned} \tag{31}$$

For instance, in the affinity system in Fig. 1, $I_{\text{aff}}(3)$ is given by 1. Furthermore, only one job will be added or removed at a discrete time event. A new event at time t_k could only violate (3) if at time t_{k-1} (3) holds with equality, i.e.,

$$\sum_{i=m}^{\infty} \overline{Q}_i^{\text{aff}}(t_{k-1}) = \sum_{i=m}^{\infty} \overline{Q}_i^{\text{ref}}(t_{k-1}) \tag{32}$$

with $m \geq 1$. Therefore, we only focus on this setting in the induction step.

Arrival. At time t_k, an arrival occurs, and first the nth ordered position is selected. The updated reference system looks as follows:

$$\overline{Q}_j^{\text{ref}}(t_k) = \begin{cases} \overline{Q}_j^{\text{ref}}(t_k^-) + 1, & \text{if } j = I_{\text{ref}}(n) + 1, \\ \overline{Q}_j^{\text{ref}}(t_k^-), & \text{otherwise.} \end{cases} \tag{33}$$

If the newly arrived job is assigned as a type-II job in the affinity system or no arrival takes place due to the coupling, (3) is trivially satisfied. We consider the setting where the job is assigned as a type-I job to a server at position $n_{\text{aff}} \leq n$, such that

$$\overline{Q}_j^{\text{aff}}(t_k) = \begin{cases} \overline{Q}_j^{\text{aff}}(t_k^-) + 1, & \text{if } j = I_{\text{aff}}(n_{\text{aff}}) + 1, \\ \overline{Q}_j^{\text{aff}}(t_k^-), & \text{otherwise.} \end{cases} \tag{34}$$

Moreover, the left-hand side of (3) remains unchanged if $m > I_{\text{aff}}(n_{\text{aff}}) + 1$ so that the order in (3) is preserved. Now, fix $m \leq I_{\text{aff}}(n_{\text{aff}}) + 1$. If we now show that also $I_{\text{ref}}(n) \geq m - 1$, then (3) remains valid since both sides are raised by one. We use (32) and the induction hypothesis for $m - 1$ at time t_k^- to obtain

$$\begin{aligned} \overline{Q}_{m-1}^{\text{aff}}(t_k^-) &= \sum_{i=m-1}^{\infty} \overline{Q}_i^{\text{aff}}(t_k^-) - \sum_{i=m}^{\infty} \overline{Q}_i^{\text{aff}}(t_k^-) \\ &\leq \sum_{i=m-1}^{\infty} \overline{Q}_i^{\text{ref}}(t_k^-) - \sum_{i=m}^{\infty} \overline{Q}_i^{\text{ref}}(t_k^-) = \overline{Q}_{m-1}^{\text{ref}}(t_k^-). \end{aligned} \tag{35}$$

Then, it follows that $I_{\text{aff}}(n_{\text{aff}}) \geq m - 1$ implies $I_{\text{ref}}(n) \geq m - 1$, which concludes the derivation if the event at time t_k is an arrival.

Departure. If at time t_k a departure takes place, one of the following four scenarios will occur:

1. There is a job completion of a type-I job in the affinity system and of a job in the reference system.
2. There is only a departure of a job in the reference system.
3. There is only a departure of a type-I job in the affinity system.
4. There is no departure of a type-I job in the affinity system or a job in the reference system.

It is clear that we only need to investigate the first two scenarios.

Scenario 1. Let P_{aff} and P_{ref} be the sets of ordered positions of servers in the affinity and reference system, respectively, that are working on (type-I) jobs just before time t_k. Define P as the intersection $P_{\text{aff}} \cap P_{\text{ref}}$. Let $n \in P$ be the position of the servers in both the affinity and the reference system from which a job will be removed. The updated states will be

$$\begin{aligned} \overline{Q}_j^{\text{aff}}(t_k) &= \begin{cases} \overline{Q}_j^{\text{aff}}(t_k^-) - 1, & \text{if } j = I_{\text{aff}}(n), \\ \overline{Q}_j^{\text{aff}}(t_k^-), & \text{otherwise,} \end{cases} \\ \overline{Q}_j^{\text{ref}}(t_k) &= \begin{cases} \overline{Q}_j^{\text{ref}}(t_k^-) - 1, & \text{if } j = I_{\text{ref}}(n), \\ \overline{Q}_j^{\text{ref}}(t_k^-), & \text{otherwise.} \end{cases} \end{aligned} \tag{36}$$

We will focus on $m \leq I_{\text{ref}}(n)$, since for $m > I_{\text{ref}}(n)$ (3) remains trivially valid. A similar argument as above will be used to show that $I_{\text{aff}}(n) \geq m$, so that both sides

will be lowered by one compared to the event time t_{k-1}. We use (32) and the induction hypothesis for $m+1$ at time t_k^- to obtain $\overline{Q}_m^{\text{aff}}(t_k^-) \geq \overline{Q}_m^{\text{ref}}(t_k^-)$. Then, it follows that $I_{\text{ref}}(n) \geq m$ implies $I_{\text{aff}}(n) \geq m$ which concludes the proof of scenario 1.
Scenario 2. Let $n \in P_{\text{ref}} \setminus P$ be the position where a job leaves the reference system. Then for all j

$$\begin{aligned} \overline{Q}_j^{\text{aff}}(t_k) &= \overline{Q}_j^{\text{aff}}(t_k^-), \\ \overline{Q}_j^{\text{ref}}(t_k) &= \begin{cases} \overline{Q}_j^{\text{ref}}(t_k^-) - 1, & \text{if } j = I_{\text{ref}}(n), \\ \overline{Q}_j^{\text{ref}}(t_k^-), & \text{otherwise.} \end{cases} \end{aligned} \tag{37}$$

Again we focus on $m \leq I_{\text{ref}}(n)$. Fixing m, we will show by contradiction that (32) cannot occur, so that (3) is preserved at time t_k since the right-hand side can be lowered by at most one. Assuming that (32) does hold and using the induction hypothesis on $m+1$, we conclude that $\overline{Q}_m^{\text{aff}}(t_k^-) \geq \overline{Q}_m^{\text{ref}}(t_k^-)$. Now,

$$\overline{Q}_m^{\text{aff}}(t_k^-) \geq \overline{Q}_m^{\text{ref}}(t_k^-) \geq \overline{Q}_{I_{\text{ref}}(n)}^{\text{ref}}(t_k^-) \geq N - n + 1 \geq |P| + 1, \tag{38}$$

since $N - |P_{\text{ref}}| < n \leq N - |P|$. This implies that $\overline{Q}_m^{\text{aff}}(t_k^-) > |P|$; however, there are only $|P| = |P_{\text{aff}}|$ servers working on a type-I job in the affinity system. This leads to a contradiction and concludes the proof of Lemma 1.

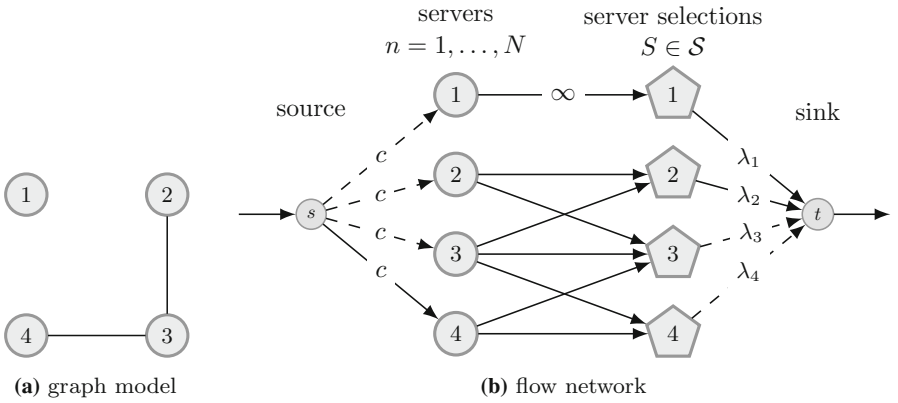
5.2 Proof: equivalent stability conditions

As mentioned in Sect. 3.2, our affinity-scheduling model without any type-II jobs is a special instance of the parallel server model with multiple job classes described by Foss and Chernova in [7]. Each job class has its own arrival rate and set of available servers with a server or class-dependent service time distribution. The JSQ policy is one of the various potential allocation policies. Our model meets this description if there is a job class for each of the different server selections $S \in \mathcal{S}$ and a common service rate μ_1 for all the servers.

In the above-described special setting, the necessary and sufficient stability condition in Theorems 2.5 and 2.7 in [7] reduces to

$$\rho_0 = \max_{J \subseteq \{1,\ldots,N\}} \left\{ \frac{1}{|J|\mu_1} \sum_{U \subseteq J} \lambda_U \right\} < 1. \tag{39}$$

With λ_U, the arrival rate to server set U if U is a server selection from $\mathcal{S}$ and zero otherwise, this stability condition implies that the aggregate arrival rate to each set of servers may never exceed the total service capacity of this set. Our stability condition, on the other hand, guarantees this for each server individually by optimizing the way the arriving streams of jobs can be divided among the servers, i.e., $\lambda_0 < \mu_1$ with λ_0 as defined in (4). This constructive representation serves the purposes in the stochastic coupling argument in Sect. 3.2.



(a) graph model **(b)** flow network

Fig. 12 **a** Graph model with four nodes and arrival rate λ_S for each of the server selections S in $\mathcal{S} = \{\{1\}, \{2, 3\}, \{2, 3, 4\}, \{3, 4\}\}$. **b** Corresponding flow network

Proposition 1 *The conditions $\rho_0 \leq 1$ and $\lambda_0 \leq \mu_1$ are equivalent to ρ_0 and λ_0 as defined in (39) and (4), respectively.*

The proof of Proposition 1 consists of two parts: first a flow network is constructed given the model parameters under consideration; then, an argument using the max-flow min-cut theorem [6] is applied to conclude the equivalence.

From a given instance of the affinity-scheduling model, a network is constructed consisting of a source node (s), a sink node (t) and nodes for each server $n \in \{1, \ldots, N\}$ and each server selection $S \in \mathcal{S}$. There are directed edges from the source s to each server node n with capacity $c \geq 0$; the value of c will be specified in the second part of the proof. Moreover, there are edges with infinite capacity from each server to the server selections that include this server. Finally, each server selection node S is connected to the sink node t with the arrival rate λ_S as its edge capacity. A visualization of the flow network construction is shown in Fig. 12.

Let $F(c)$ and $C(c)$ denote the maximum achievable flow and minimum cut that separates the source s from the sink t in the constructed network, respectively, for a given capacity c. Note that

$$F(c) \leq \sum_{S \in \mathcal{S}} \lambda_S \tag{40}$$

since the edges (S, t), $S \in \mathcal{S}$, form a cut, with equality once $c \geq \lambda_0$ due to the definition of λ_0. Furthermore, the edges connecting server nodes and server selection nodes will never be included in the minimum cut, and the minimum cut has a specific form. Namely, if J_c is a subset of $\{1, \ldots, N\}$, then the cut consists of all edges (s, n) with $n \in J_c$ and the edges (S, t) with $S \in \mathcal{S}$ such that S is not included in J_c. Indeed, note that given the set J_c, all edges (S, t) for which S is not included in J_c must be captured in order to have a cut, but any edges (S, t) for which $S \subseteq J_c$ can be omitted while preserving the cut property. This implies that the minimum cut optimization problem can be rewritten as

$$\tilde{C}(c) = \min_{J \subseteq \{1,\ldots,N\}} \left\{ c|J| + \sum_{S \not\subseteq J} \lambda_S \right\} \tag{41}$$

without changing the value of the objective function. An example of a cut with $J = \{1, 2, 3\}$ is presented in Fig. 12b by dashed lines. Moreover, due to the max-flow min-cut theorem, it is known that

$$F(c) = C(c) = \tilde{C}(c). \tag{42}$$

If $\lambda_0 \leq \mu_1$, then let $\lambda_0 \leq c \leq \mu_1$. Using (40) and (41) together with (42) leads to

$$\begin{aligned} &\sum_{S \in \mathcal{S}} \lambda_S \leq c|J| + \sum_{S \not\subseteq J} \lambda_S \\ &\Leftrightarrow \sum_{S \subseteq J} \lambda_S \leq c|J| \end{aligned} \tag{43}$$

for all $J \subseteq \{1, \ldots, N\}$. Hence, $\rho_0 \leq 1$.

If $\rho_0 \leq 1$, then the equivalence in (43) holds for all $J \subseteq \{1, \ldots, N\}$ and $c = \mu_1$. This implies in particular that

$$\sum_{S \in \mathcal{S}} \lambda_S \leq \tilde{C}(\mu_1) = F(\mu_1) \leq \sum_{S \in \mathcal{S}} \lambda_S. \tag{44}$$

By construction, λ_0 is the smallest possible value for c such that (40) holds with equality. So, $\lambda_0 \leq \mu_1$.

5.3 Proofs: fluid limit and fixed point analysis

5.3.1 Derivation of fluid limit (15)

First, consider the stochastic process with N servers and its corresponding flow conservation equations. Next, the martingale methods as outlined by Pang et al. [21] and Brémaud [3] are applied and the limit as N tends to infinity of the fluid-scaled process is studied. Then, (15) is obtained from the resulting system of integral equations.

Step 1: flow conservation equations Let $p_{ij}^N(q, t)$ be the probability that an arriving job at time t is assigned to a server with i type-I jobs and j type-II jobs as a type-q job, with $q \in \{\mathrm{I}, \mathrm{II}\}$. As before, we will omit the time dependence t to ease the notation.

We can only assign a job as a type-I or type-II job to an idle server; assignments to servers with a higher configuration will always take place as a type-I job. The corresponding transition probabilities are given by

$$p_{00}^N(\mathrm{I}) = 1 - \left(1 - \frac{Q_{00}^N}{N}\right)^d, \tag{45}$$

the probability that an idle server is present in the primary selection, and

$$p_{00}^N(\mathrm{II}) = \mathbb{1}\{Q_{00}^N > 0\}\left(1 - \frac{Q_{00}^N}{N}\right)^d, \tag{46}$$

the probability that the primary selection does not contain an idle server while they are present. As mentioned in the model description, the secondary selection contains all servers that are not in the primary selection. Hence, the indicator function $\mathbb{1}\{Q_{00}^N > 0\}$ emerges in the probabilities.

An arriving job will be assigned as a type-I job to a server with configuration $(i, 0)$, with $i \geq 1$, if the minimum configuration in the primary selection is given by $(i, 0)$ and when there are no completely idle servers that can be included in the secondary selection. The corresponding probability is given by the probability that the primary selection contains only servers with at least i type-I jobs minus the probability that all d servers have a configuration strictly higher than $(i, 0)$. Thus, for $i \geq 1$,

$$p_{i0}^N(\mathrm{I}) = \mathbb{1}\{Q_{00}^N = 0\}\left[\left(\sum_{k\geq i}\left[\frac{Q_{k0}^N}{N} + \frac{Q_{k1}^N}{N}\right]\right)^d - \left(\frac{Q_{i1}^N}{N} + \sum_{k\geq i+1}\left[\frac{Q_{k0}^N}{N} + \frac{Q_{k1}^N}{N}\right]\right)^d\right]. \tag{47}$$

In a similar way, we obtain $p_{i1}^N(\mathrm{I})$, for $i \geq 0$:

$$p_{i1}^N(\mathrm{I}) = \mathbb{1}\{Q_{00}^N = 0\}\left[\left(\frac{Q_{i1}^N}{N} + \sum_{k\geq i+1}\left[\frac{Q_{k0}^N}{N} + \frac{Q_{k1}^N}{N}\right]\right)^d - \left(\sum_{k\geq i+1}\left[\frac{Q_{k0}^N}{N} + \frac{Q_{k1}^N}{N}\right]\right)^d\right]. \tag{48}$$

Once these probabilities are set, the flow conservation equations can be constructed. The randomness in the stochastic model is caused by Poisson arrivals and exponentially distributed service times, so that the number of arrivals and service completions can be counted using Poisson processes with appropriately chosen rates. Define a set of independent Poisson processes with rate 1. Let $P_{A_{00,q}}$ denote the Poisson counting process for the number of arriving type-q jobs at servers with configuration $(0, 0)$, and $P_{A_{ij}}$, $i + j \geq 1$, reflects the number of arriving jobs at servers with configuration (i, j). Similarly, define the counting process of the service completions $P_{S_{ij}}$, $i+j \geq 1$. Furthermore, if $i \geq 1$, the number of servers at time t with at least i type-I jobs and exactly j type-II jobs depends on its initial state $(\overline{Q}_{ij}^N(0))$, the number of service completions of jobs at servers with configuration (i, j) and the number of arrivals at servers in configuration $(i-1, j)$ within the time interval $[0, t)$. We obtain the following flow conservation equations for the stochastic model $(\overline{Q}_{ij}^N)_{i,j}$ with N servers and total arrival rate λN. Let $i \geq 2$:

$$
\begin{aligned}
\overline{Q}^N_{00}(t) &= \overline{Q}^N_{00}(0) + P_{S_{01}}\left(\mu_2 \int_0^t Q^N_{01}(s)\,\mathrm{d}s\right) \\
&\quad - P_{A_{00}}\left(\lambda N \int_0^t \left[p^N_{00}(\mathrm{I}, s) + p^N_{00}(\mathrm{II}, s)\right]\mathrm{d}s\right), \\
\overline{Q}^N_{01}(t) &= \overline{Q}^N_{01}(0) - P_{S_{01}}\left(\mu_2 \int_0^t Q^N_{01}(s)\,\mathrm{d}s\right) + P_{A_{00,\mathrm{II}}}\left(\lambda N \int_0^t p^N_{00}(\mathrm{II}, s)\,\mathrm{d}s\right), \\
\overline{Q}^N_{10}(t) &= \overline{Q}^N_{10}(0) - P_{S_{10}}\left(\mu_1 \int_0^t Q^N_{10}(s)\,\mathrm{d}s\right) + P_{A_{00,\mathrm{I}}}\left(\lambda N \int_0^t p^N_{00}(\mathrm{I}, s)\,\mathrm{d}s\right), \\
\overline{Q}^N_{11}(t) &= \overline{Q}^N_{11}(0) - P_{S_{11}}\left(\mu_1 \int_0^t Q^N_{11}(s)\,\mathrm{d}s\right) + P_{A_{01}}\left(\lambda N \int_0^t p^N_{01}(\mathrm{I}, s)\,\mathrm{d}s\right), \\
\overline{Q}^N_{i0}(t) &= \overline{Q}^N_{i0}(0) - P_{S_{i0}}\left(\mu_1 \int_0^t Q^N_{i0}(s)\,\mathrm{d}s\right) + P_{A_{i-1,0}}\left(\lambda N \int_0^t p^N_{i-1,0}(\mathrm{I}, s)\,\mathrm{d}s\right), \\
\overline{Q}^N_{i1}(t) &= \overline{Q}^N_{i1}(0) - P_{S_{i1}}\left(\mu_1 \int_0^t Q^N_{i1}(s)\,\mathrm{d}s\right) + P_{A_{i-1,1}}\left(\lambda N \int_0^t p^N_{i-1,1}(\mathrm{I}, s)\,\mathrm{d}s\right).
\end{aligned} \tag{49}
$$

Due to the Poisson split property, we define $P_{A_{00}}$ as the sum of the two processes $P_{A_{00,\mathrm{I}}}$ and $P_{A_{00,\mathrm{II}}}$.

Step 2: Fluid-scaled process Dividing both sides of the equations by N results in a fluid-scaled process. Further, because of the martingale results in [3] and [21], we can define noise terms $e_{ij}(N)$ that tend to 0 as $N \to \infty$ with $i \geq 0$ and $j \in \{0, 1\}$. The fluid-scaled system can be rewritten as follows, for $i \geq 2$:

$$
\begin{aligned}
\frac{\overline{Q}^N_{00}(t)}{N} &= \frac{\overline{Q}^N_{00}(0)}{N} + \mu_2 \int_0^t \frac{Q^N_{01}(s)}{N}\,\mathrm{d}s \\
&\quad - \lambda \int_0^t \left[p^N_{00}(\mathrm{I}, s) + p^N_{00}(\mathrm{II}, s)\right]\mathrm{d}s + e_{00}(N), \\
\frac{\overline{Q}^N_{01}(t)}{N} &= \frac{\overline{Q}^N_{01}(0)}{N} - \mu_2 \int_0^t \frac{Q^N_{01}(s)}{N}\,\mathrm{d}s + \lambda \int_0^t p^N_{00}(\mathrm{II}, s)\,\mathrm{d}s + e_{01}(N), \\
\frac{\overline{Q}^N_{10}(t)}{N} &= \frac{\overline{Q}^N_{10}(0)}{N} - \mu_1 \int_0^t \frac{Q^N_{10}(s)}{N}\,\mathrm{d}s + \lambda \int_0^t p^N_{00}(\mathrm{I}, s)\,\mathrm{d}s + e_{10}(N),
\end{aligned}
$$

$$\frac{\overline{Q}^N_{11}(t)}{N} = \frac{\overline{Q}^N_{11}(0)}{N} - \mu_1 \int_0^t \frac{Q^N_{11}(s)}{N}\,\mathrm{d}s + \lambda \int_0^t p^N_{01}(\mathrm{I}, s)\,\mathrm{d}s + e_{11}(N),$$

$$\frac{\overline{Q}^N_{i0}(t)}{N} = \frac{\overline{Q}^N_{i0}(0)}{N} - \mu_1 \int_0^t \frac{Q^N_{i0}(s)}{N}\,\mathrm{d}s + \lambda \int_0^t p^N_{i-1,0}(\mathrm{I}, s)\,\mathrm{d}s + e_{i0}(N),$$

$$\frac{\overline{Q}^N_{i1}(t)}{N} = \frac{\overline{Q}^N_{i1}(0)}{N} - \mu_1 \int_0^t \frac{Q^N_{i1}(s)}{N}\,\mathrm{d}s + \lambda \int_0^t p^N_{i-1,1}(\mathrm{I}, s)\,\mathrm{d}s + e_{i1}(N). \quad (50)$$

Step 3: Toward fluid limits While making the transition from integral equations to differential equations with N tending to infinity, the representation of the departure terms in (15) is straightforward. The arrival terms in the differential equations, on the other hand, are not immediately obvious.

To illustrate the difficulty, assume there are among the N servers only a small number of idle servers. As the assignment policy describes, one of these servers will be selected by an arriving job. If the number of idle servers is small and the arrival rate is sufficiently high, rapid switches will occur in the indicator function $\mathbb{1}\{Q^N_{00} = 0\}$. A server that becomes idle due to a service completion will immediately be selected again by the arriving job. However, the fraction of idle servers (Q^N_{00}/N) will be more robust against these changes due to the fluid scaling.

In general, this phenomenon is called 'separation of time scales' as described by Hunt and Kurtz [14]. One observes the interaction of two processes. One process evolves very fast, namely the number of idle servers, while the second process evolves much slower, the occupancy fractions in this setting. In order to obtain the arrival terms of the fluid limit, we should be able to combine these processes. Focusing on the first arrival integral in (50), the question arises of how to handle the expression

$$\lim_{N\to\infty} \lambda \int_0^t \left[p^N_{00}(\mathrm{I}, s) + p^N_{00}(\mathrm{II}, s) \right] \mathrm{d}s = \lim_{N\to\infty} \lambda \int_0^t \mathbb{1}\{Q^N_{00}(s) > 0\}\,\mathrm{d}s. \quad (51)$$

A similar problem is analyzed in [14], where one needs to take the limit of a integral of an indicator function. The existence of a measure α is deduced such that

$$\lim_{N\to\infty} \lambda \int_0^t \mathbb{1}\{Q^N_{00}(s) > 0\}\,\mathrm{d}s = \lambda \int_0^t \alpha(s)\,\mathrm{d}s. \quad (52)$$

The existence of this function α, which does not need to be continuous, can be justified by the following reasoning: In a small time interval, say $[0, \delta t]$, the number of idle servers is a heavily fluctuating process, though the process describing the occupancy fractions is approximately constant. During this small interval, the number of idle servers can be considered as a birth-and-death process with 'death' rate λ, since an arriving job causes a reduction in the number of idle servers. The 'birth' rate is determined by the occupancy fractions, i.e., the fraction of servers that is working on type-I or type-II jobs. Then, it is argued in [14] that

$$\frac{1}{\delta t}\int_0^{\delta t} \mathbb{1}\{Q_{00}^N(s) > 0\}\,\mathrm{d}s, \tag{53}$$

after application of the ergodic theorem, converges to an invariant measure if N tends to infinity. This invariant measure will give rise to the function α. One already senses that the presence or absence of idle servers should be handled as two different cases. Therefore, we make a distinction between q_{00} strictly positive or equal to zero in the intuitive explanation of the structure of the fluid limit.

The case $q_{00} > 0$. When the number of idle servers is sufficiently large, each arriving job will be assigned to an idle server for sure. A fraction

$$(1 - q_{00})^d \tag{54}$$

of the arriving jobs will be assigned as type-II jobs, which causes the changes in (15) for $\overline{q}_{00}$, $\overline{q}_{01}$ and $\overline{q}_{10}$.

The case $q_{00} = 0$. Idle servers are generated at rate $\mu_1 q_{10} + \mu_2 q_{01}$. Since d is finite, the probability that the primary selection would contain an idle server is negligible; each idle server will be provided with a type-II job when the arrival rate is high enough. If $\tilde{\lambda} = (\lambda - \mu_1 q_{10} + \mu_2 q_{01})^+$ is strictly larger than zero, a fraction

$$\frac{\mu_1 q_{10} + \mu_2 q_{01}}{\lambda} = \frac{\lambda - \tilde{\lambda}}{\lambda} \tag{55}$$

of the stream of incoming jobs will immediately be redirected to the idle servers as a type-II job. The excess stream of incoming jobs (fraction $\tilde{\lambda}/\lambda$) will not observe any idle server and will start to form (type-I) queues in front of the servers of the primary selection according to a straightforward generalization of the transition probabilities mentioned in step 1.

This concludes the derivation of the fluid limit (15).

5.3.2 Proof of Theorem 4: fixed points

We will start with the proof of the closed-form fixed point and show that this is the only fixed point without idle servers on the fluid level, i.e., $q_{00} = 0$. Next, we will consider fixed points with $q_{00} > 0$.

Fixed points with $q_{00} = 0$. The correctness of the expression in (16) can easily be confirmed by substitution into (15). The result can be established in two steps. First, we observe that the derivatives of $(\overline{q}_{i0})_i$ in (15) remain zero once $(\overline{q}^*_{i0})_i$ equals zero. Then, we substitute $(\overline{q}^*_{i0})_i = 0$ into the derivatives of $(\overline{q}_{i1})_i$. For $i \geq 1$, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{q}^*_{i1} = \mu_1(\overline{q}^*_{i+1,1} - \overline{q}^*_{i1}) + \tilde{\lambda}\left[(\overline{q}^*_{i-1,1})^d - (\overline{q}^*_{i1})^d\right] = 0. \tag{56}$$

These equations can be solved, and one obtains the fixed point as given in (16). Note the similarity between (56) and the fluid limit of a JSQ(d) policy with reduced arrival

rate

$$\tilde{\lambda} = \lambda - \frac{\mu_1 - \lambda}{\mu_1 - \mu_2}\mu_2 = \lambda - \mu_2 q_{01}^*, \tag{57}$$

in a setting where each of the exchangeable servers works at rate μ_1 [18].

Second, this fixed point is unique under the condition that q_{00} equals zero. From Lemma 2 in [18], we know that the fixed point of the fluid limit in the JSQ(d) setting is unique when $d \geq 2$. This implies that under the condition that all servers have a type-II job, i.e., $\overline{q}_{i0}^* = 0$ for all i, uniqueness is guaranteed. Assume by contradiction that another fixed point exists without idle servers but with possibly a positive cumulative fraction $\overline{q}_{i0}^*$ for some i. We focus on the differential equations of $(\overline{q}_{i0})_{i\geq 1}$ under this fixed point. From

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{10}^* = \mu_1(\overline{q}_{20}^* - \overline{q}_{10}^*) = 0, \tag{58}$$

we get that $\overline{q}_{10}^* = \overline{q}_{20}^*$. Repeating this procedure for $i = 2$,

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{20}^* &= \mu_1(\overline{q}_{30}^* - \overline{q}_{20}^*) + \tilde{\lambda}\left[(\overline{q}_{10}^* - \overline{q}_{11}^*)^d - (\overline{q}_{20}^* - \overline{q}_{11}^*)^d\right] \\ &= \mu_1(\overline{q}_{30}^* - \overline{q}_{20}^*) = 0,\end{aligned} \tag{59}$$

resulting in $\overline{q}_{20}^* = \overline{q}_{30}^*$. By induction, we could show that $\overline{q}_{i0}^* = \overline{q}_{i+1,0}^*$; for $i \geq 1$, this leads to $\overline{q}_{i0}^* = 0$ for $i \geq 1$. This proves the uniqueness of the fixed point when q_{00} equals zero.

Fixed points with $q_{00} > 0$. Under this setting, the fluid limit equations (15) simplify significantly:

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{00} = \mu_2 q_{01} - \lambda(1 - q_{00})^d, \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{01} = -\mu_2 q_{01} + \lambda(1 - q_{00})^d, \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{10} = -\mu_1 q_{10} + \lambda\left(1 - (1 - q_{00})^d\right), \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{11} = -\mu_1 q_{11}, \\ \text{for } i \geq 2, \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{i0} = -\mu_1 q_{i0}, \\ \frac{\mathrm{d}}{\mathrm{d}t}\overline{q}_{i1} = -\mu_1 q_{i1}. \end{cases} \tag{60}$$

For any fixed point, it should hold that $(\overline{q}_{i0}^*)_{i\geq 2} = 0$ and $(\overline{q}_{i1}^*)_{i\geq 1} = 0$. This implies that the only positive fractions are q_{00}, q_{01} and q_{10}. The resulting system of differential equations is given by

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}q_{00} = \mu_1 q_{10} + \mu_2 q_{01} - \lambda, \\ \frac{\mathrm{d}}{\mathrm{d}t}q_{01} = -\mu_2 q_{01} + \lambda(1 - q_{00})^d, \\ \frac{\mathrm{d}}{\mathrm{d}t}q_{10} = -\mu_1 q_{10} + \lambda\left(1 - (1 - q_{00})^d\right). \end{cases} \tag{61}$$

From the second and third equality, it is clear that once q^*_{00} is known, we know the entire fixed point:

$$\begin{cases} q^*_{01} = \frac{\lambda}{\mu_2}(1 - q^*_{00})^d, \\ q^*_{10} = \frac{\lambda}{\mu_1}\left(1 - (1 - q^*_{00})^d\right). \end{cases} \tag{62}$$

The system in (61) is linearly dependent. We use the fact that q_{00}, q_{01} and q_{10} must sum up to one to determine q_{00}. It must hold that

$$1 = q_{00} + (1 - q_{00})^d \left(\frac{\lambda}{\mu_2} - \frac{\lambda}{\mu_1}\right) + \frac{\lambda}{\mu_1}.$$

Define $x \doteq 1 - q_{00}$. We are interested in the zero points of the polynomial f within $[0, 1)$, with

$$f(x) = x^d \left(\frac{\lambda}{\mu_2} - \frac{\lambda}{\mu_1}\right) - x + \frac{\lambda}{\mu_1}. \tag{63}$$

We will evaluate the existence of the fixed points based on the behavior of f and its derivative,

$$f'(x) = d \left(\frac{\lambda}{\mu_2} - \frac{\lambda}{\mu_1}\right) x^{d-1} - 1. \tag{64}$$

Furthermore,

$$\begin{aligned} f(0) &= \frac{\lambda}{\mu_1} > 0, \\ f(1) &= \frac{\lambda}{\mu_2} - 1 > 0, \end{aligned} \tag{65}$$

and f' is monotone increasing on $(0, 1)$ with

$$\begin{aligned} f'(0) &= -1 < 0, \\ f'(1) &= d\lambda \left(\frac{1}{\mu_2} - \frac{1}{\mu_1}\right) - 1. \end{aligned} \tag{66}$$

Since f is positive in both its endpoints and the derivative f' is monotone increasing, we need at least a vanishing derivative in $(0, 1)$ in order to have a fixed point. This is guaranteed when $f'(1) > 0$; this is the first condition from (17). We now know that f attains a local minimum at

$$\tilde{x} \doteq \left(\frac{1}{d} \frac{1}{\lambda \left(\frac{1}{\mu_2} - \frac{1}{\mu_1}\right)}\right)^{\frac{1}{d-1}} \tag{67}$$

and is strictly positive in its endpoints. If $f(\tilde{x})$ is exactly zero, we have one fixed point, namely $q^*_{00} = 1 - \tilde{x}$. But only in very special cases the second condition of (17) is

satisfied with equality for an arbitrary choice of d_1, λ, μ_1 and μ_2. On the other hand, if $f(\tilde{x}) < 0$, i.e., if also

$$\left(1-\frac{1}{d}\right)\left(\frac{1}{d}\right)^{\frac{1}{d-1}} > \frac{\lambda}{\mu_1}\left(\lambda\left(\frac{1}{\mu_2}-\frac{1}{\mu_1}\right)\right)^{\frac{1}{d-1}} \tag{68}$$

holds, then we have exactly two fixed points such that $q_{00}+q_{01}+q_{10}=1$. There is one fixed point situated at each side of $\tilde{x}$ in the interval $(0, 1)$. This gives that for d large enough we can find two solutions of the reduced system of differential equations. It can be shown by contradiction that both fixed points are larger than λ/μ_1, so the corresponding fractions of idle servers are smaller than $1-\lambda/\mu_1$.

For completeness, we mention that $\lambda = \mu_2$ would imply that $f(1) = 0$ and so the proportion of idle servers is zero, which violates the assumption that $q_{00} > 0$. Moreover, if $\lambda < \mu_2$, then the polynomial f vanishes in the interval $(0, 1)$. The monotone increasing property of the derivative of f leads to the fact that there exists a unique fixed point x^* in $(0, 1)$. This results in a unique fixed point $(q_{00}^*, q_{01}^*, q_{10}^*)$ with $q_{00}^* > 0$.

This concludes the proof of Theorem 4.

5.3.3 Proof of Theorem 5: local (in)stability

We will prove local (in)stability using the indirect Lyapunov method based on the Hartman–Grobman theorem [13]. This theorem states that a system of differential equations behaves near its fixed points as its linearized version. The eigenvalues of the linearized system will define the local behavior of the system unless one of the eigenvalues has a real part equal to zero; then, the Hartman–Grobman theorem is inconclusive. If we were to immediately apply this theorem to one of the two fixed points of (61), we would obtain an eigenvalue exactly equal to zero, but one can resolve this issue since (61) is a redundant system. Since $q_{00}+q_{01}+q_{10}=1$, it is sufficient to know the instantaneous change of two variables. Each elimination will lead to the same two eigenvalues so we can remove, for instance, the third equation from (61):

$$\begin{cases} \frac{\mathrm{d}}{\mathrm{d}t}q_{00} = \mu_1(1-q_{00}-q_{01}) + \mu_2 q_{01} - \lambda, \\ \frac{\mathrm{d}}{\mathrm{d}t}q_{01} = -\mu_2 q_{01} + \lambda(1-q_{00})^d. \end{cases} \tag{69}$$

Let $(q_{00}^*, q_{01}^*, q_{10}^*)$ denote a fixed point, then the matrix of the linearized system looks as follows near its fixed point:

$$\begin{bmatrix} -\mu_1 & \mu_2-\mu_1 \\ -\lambda d_1(1-q_{00}^*)^{d-1} & -\mu_2 \end{bmatrix}. \tag{70}$$

The corresponding eigenvalues are given by

$$\alpha_\pm = \frac{1}{2}\left[-(\mu_1+\mu_2) \pm \sqrt{(\mu_1-\mu_2)^2 + 4\lambda d(\mu_1-\mu_2)(1-q_{00}^*)^{d-1}}\right]. \tag{71}$$

Since $\mu_1 > \mu_2$, the quantity under the root is always positive, so the square root is real. This implies, furthermore, that $\alpha_- < 0$. To determine the sign of α_+, we need to make a distinction between the two fixed points. From the proof of Theorem 4, we know that the two fixed points are on both sides of $\tilde{x}$, with $\tilde{x}$ as in (67). For

$$1 - q_{00}^* > \tilde{x} = \left(\frac{1}{d}\frac{\mu_1\mu_2}{\lambda(\mu_1 - \mu_2)}\right)^{\frac{1}{d-1}}, \tag{72}$$

we have that

$$\begin{aligned} 2\alpha_+ &> -(\mu_1 + \mu_2) + \sqrt{(\mu_1 - \mu_2)^2 + 4\lambda d(\mu_1 - \mu_2)\left(\frac{1}{d}\frac{\mu_1\mu_2}{\lambda(\mu_1 - \mu_2)}\right)} \\ &= -(\mu_1 + \mu_2) + \sqrt{(\mu_1 + \mu_2)^2} \\ &= 0. \end{aligned} \tag{73}$$

This shows that the fixed point with the smallest fraction of idle servers is unstable.

When $1 - q_{00}^* < \tilde{x}$, it follows in a similar way that $2\alpha_+ < 0$. This shows that the fixed point with the largest proportion of idle servers is locally stable. This concludes the proof of Theorem 5.

6 Conclusion and outlook

We investigated load balancing issues in a service system where particular servers are better equipped to process certain jobs due to affinity or compatibility relations. The general model in particular covers the setting with an underlying network topology G_N, referred to as the *neighborhood model*. The analysis of the neighborhood model is severely complicated by the lack of exchangeability among the servers; a feature present in the supermarket modeling framework that allows mean field techniques. We constructed the novel *R-coupling*, or *restructure coupling*, to obtain stochastic performance bounds for the general model and more specific settings, for instance model instances where the underlying graph topology G_N has a specific minimum degree or is a d-regular graph.

Another instance of the general model, the *combinatorial model*, has enough inherent symmetry to conduct a fluid limit analysis. The fluid limit was stated in terms of a set of discontinuous differential equations, and its fixed point sensitively depends on the size d of the primary selection. When d is sufficiently small, a unique fixed point exists, but the associated waiting time does not vanish. When the primary selection is sufficiently large, a fixed point arises that does provide a zero waiting time. On the other hand, the above-mentioned fixed point still persists, giving rise to bistability issues.

As mentioned above, the stochastic upper bounds for the neighborhood model in terms of a supermarket model with a JSQ(d) policy require the degrees in the underlying graph to be relatively high compared to d. To some extent, this indicates that the performance may be poor in certain pathological cases even when the node

degrees are fairly high. An interesting topic for further research would be to extend the R-coupling and possibly identify relevant structural conditions on the graph topology in order to sharpen these bounds.

For a fixed set of model parameters with a uniform arrival rate per server selection and the described affinity-scheduling policy, it seems plausible to expect an improvement in performance when the set of server selections $\mathcal{S}$ grows bigger and becomes more diverse in some appropriate sense. This would imply that the performance of the combinatorial model provides a conservative estimate for the performance of, for instance, the neighborhood model with an underlying regular graph structure. Moreover, recall that a supermarket model with a JSQ(d) policy is equivalent to the combinatorial model with server selection of size d when jobs cannot be assigned as type-II jobs, which effectively occurs when μ_2 approaches zero. This suggests that, for sufficiently small μ_2, the supermarket model with a JSQ(d) policy provides stochastic lower bounds for our affinity-scheduling model with server selections of size at most d.

The bistability of the fluid limit of the combinatorial model for large values of d not only precludes any convergence statements for the stationary distribution, but also suggests that the assignment strategy could possibly be refined. In future work we intend to examine such refinements and establish that these eliminate the *queueing* fixed point and render the *no-queueing* fixed point globally stable.

References

1. Bean, N., Gibbens, R., Zachary, S.: Dynamic and equilibrium behavior of controlled loss networks. Ann. Appl. Probab. **7**, 873–885 (1997)
2. Benaïm, M., Le Boudec, J.Y.: On mean field convergence and stationary regime. arXiv preprint arXiv:1111.5710 (2011)
3. Brémaud, P.: Point Processes and Queues, Martingale Dynamics. Springer, New York (1981)
4. Devroye, L.: Non-uniform Random Variate Generation. Springer, New York (1986)
5. Ephremides, A., Varaiya, P., Walrand, J.: A simple dynamic routing problem. IEEE Trans. Autom. Control **25**(4), 690–693 (1980)
6. Ford, L.R., Fulkerson, D.R.: Flows in Networks. Princeton University Press, Princeton (2015)
7. Foss, S., Chernova, N.: On the stability of a partially accessible multi-station queue with state-dependent routing. Queueing Syst. **29**(1), 55–73 (1998)
8. Gamarnik, D., Tsitsiklis, J.N., Zubeldia, M.: Delay, memory, and messaging tradeoffs in distributed service systems. ACM SIGMETRICS Perform. Eval. Rev. **44**(1), 1–12 (2016)
9. Gardner, K., Stephens, C.: Smart dispatching in heterogeneous systems. In: Workshop on Mathematical Performance Modeling and Analysis. Phoenix, AZ (2019)
10. Gast, N.: The power of two choices on graphs: the pair-approximation is accurate? ACM SIGMETRICS Perform. Eval. Rev. **43**(2), 69–71 (2015)
11. Gibbens, R., Hunt, P., Kelly, F.: Bistability in communication networks. Disorder in physical systems pp. 113–128 (1990)
12. Harrison, J.M., López, M.J.: Heavy traffic resource pooling in parallel-server systems. Queueing Syst. **33**(4), 339–368 (1999)

13. Hartman, P.: On local homeomorphisms of Euclidean spaces. Bol. Soc. Mat. Mexicana **5**(2), 220–241 (1960)
14. Hunt, P., Kurtz, T.: Large loss networks. Stoch. Process. Appl. **53**(2), 363–378 (1994)
15. Liu, Z., Nain, P., Towsley, D.: Sample path methods in the control of queues. Queueing Syst. **21**(3–4), 293–335 (1995)
16. Lu, Y., Xie, Q., Kliot, G., Geller, A., Larus, J.R., Greenberg, A.: Join-idle-queue: a novel load balancing algorithm for dynamically scalable web services. Perform. Eval. **68**(11), 1056–1071 (2011)
17. Martirosyan, D., Robert, P.: The equilibrium states of large networks of Erlang queues. arXiv preprint arXiv:1811.04763 (2018)
18. Mitzenmacher, M.: The power of two choices in randomized load balancing. IEEE Trans. Parallel Distrib. Syst. **12**(10), 1094–1104 (2001)
19. Mukherjee, D., Borst, S.C., Van Leeuwaarden, J.S.H.: Asymptotically optimal load balancing topologies. Proc. ACM Meas. Anal. Comput. Syst. **2**(1), 14 (2018)
20. Mukherjee, D., Borst, S.C., Van Leeuwaarden, J.S.H., Whiting, P.A.: Universality of power-of-d load balancing in many-server systems. Stoch. Syst. **8**(4), 265–292 (2018). https://doi.org/10.1287/stsy.2018.0016
21. Pang, G., Talreja, R., Whitt, W.: Martingale proofs of many-server heavy-traffic limits for Markovian queues. Probab. Surv. **4**, 193–267 (2007)
22. Sparaggis, P.D., Towsley, D., Cassandras, C.G.: Sample path criteria for weak majorization. Adv. Appl. Probab. **26**(1), 155–171 (1994)
23. Stolyar, A.L.: Optimal routing in output-queued flexible server systems. Probab. Eng. Inf. Sci. **19**(2), 141–189 (2005)
24. Stoyan, D., Daley, D.J.: Comparison Methods for Queues and Other Stochastic Models. Wiley, New York (1983)
25. Turner, S.R.: The effect of increasing routing choice on resource pooling. Probab. Eng. Inf. Sci. **12**(1), 109–124 (1998)
26. Van der Boor, M., Borst, S.C., Van Leeuwaarden, J.S.H., Mukherjee, D.: Scalable load balancing in networked systems: Universality properties and stochastic coupling methods. arXiv preprint arXiv:1712.08555 (2017)
27. Vvedenskaya, N.D., Dobrushin, R.L., Karpelevich, F.I.: Queueing system with selection of the shortest of two queues: an asymptotic approach. Problemy Peredachi Informatsii **32**(1), 20–34 (1996)
28. Winston, W.: Optimality of the shortest line discipline. J. Appl. Probab. **14**(1), 181–189 (1977)
29. Yekkehkhany, A., Hojjati, A., Hajiesmaili, M.H.: GB-pandas: throughput and heavy-traffic optimality analysis for affinity scheduling. ACM SIGMETRICS Perform. Eval. Rev. **45**(2), 2–14 (2018)
30. Zachary, S., Ziedins, I.: Loss networks. In: Queueing Networks, pp. 701–728. Springer (2011)