Optimal Routing in Parallel, non-Observable Queues and the Price of Anarchy Revisited

Jonatha Anselmi[1,2] and Bruno Gaujal[1]
[1]INRIA and LIG, MESCAL project, MontBonnot Saint-Martin, 38330, France
[2]BCAM - Basque Center for Applied Mathematics, Bizkaia Technology Park, Derio, 48170, Spain
{jonatha.anselmi,bruno.gaujal}@imag.fr

***Abstract*—We consider a network of parallel, non-observable queues and analyze the "price of anarchy", an index measuring the worst-case performance loss of a decentralized system with respect to its centralized counterpart in presence of non-cooperative users. Our analysis is undertaken from the new point of view where the router has the memory of previous dispatching choices, which significantly complicates the nature of the problem. In the regime where the demands proportionally grow with the network capacity, we provide a tight lower bound on the socially-optimal response time and a tight upper bound on the price of anarchy by means of convex programming. Then, we exploit this result to show, by simulation, that the billiard routing scheme yields a response time which is remarkably close to our lower bound, implying that billiards minimize response time. To study the added-value of non-Bernoulli routers, we introduce the "price of forgetting" and prove that it is bounded from above by two, which is tight in heavy-traffic. Finally, other structural properties are derived numerically for the price of forgetting. These claim that the benefit of having memory in the router is independent of the network size and heterogeneity, while monotonically depending on the network load only. These properties yield simple product-forms well-approximating the socially-optimal response time.** [1]

I. Introduction

The "Price of Anarchy" (PoA) [21] is an index measuring the inefficiency of decentralized systems with respect to their centralized counterparts to tradeoff, in service networks, among performance, scalability, and reliability. It is defined as the worst-case response-time ratio between the situation where non-cooperative jobs (or users) behave selfishly to maximize their own individual benefit, yielding a *Nash equilibrium*, and the contrasting situation where jobs are controlled optimally by a central authority, yielding the *social optimum* or *social welfare*. While the former identifies the equilibrium point for which any unilateral deviation of each job strategy does not lower its delay, the latter represents the optimal strategy for all jobs in a centralized setting.

The interest for the PoA in the context of queueing models is currently growing because of its large spectrum of applications: Network routing, load balancing, peer-to-peer and content delivery networks, wireless networks, server farms, grid computing clusters, desktop-grid computing, and database systems; see [23], [22], [19], [13], [26], [17], [9], [1], [4], [6], [24]. The great majority of these works provide mathematical tools for characterizing and computing the mean response times in both the situations above and try to relate the PoA to the network size in different settings. This lets designers quantitatively evaluate the loss of performance when shifting to decentralized solutions (yielding Nash equilibria) and subsequently perform a suitable dimensioning of the system. In [23] it is shown that the PoA is independent of the network topology as long as the mean job arrival rate is less than the mean service rate of the slowest server, and, in this light-load regime, an upper bound is provided. When heterogeneous processor-sharing queues are considered, it is shown in [17], [26] that the PoA scales linearly with the network size, and it can only depend on the heterogeneity degree of the queues provided that these adopt the shortest-remaining-processing-time scheduling discipline [9]. In the case of multiple central authorities, e.g., the case of large server farms, the PoA is lower bounded by the square root of the number of authorities [4].

We observe that a key point common to all the above works is that the central authority, which in the remainder of the paper we refer to as *router*, achieves the social optimum in a Bernoulli setting, making its routing decisions independent each other, i.e., with no memory. In fact, the social optimum is commonly searched among all the possible Bernoulli policies through a non-linear optimization problem. In several cases, this restriction is known to yield tractable formulas for mean response times. In a more realistic framework, however, routers can exploit local information about their previous decisions and, in this context, the optimal jobs inter-arrival times of each queue are not even i.i.d. [16]. Except for special cases, this notoriously complicates the nature of the problem: The assessment of the routing policy which minimizes the mean response time as well as the analysis of such response time are current open problems; see, e.g., [16], [5], [10], [14] for the case of parallel and non-observable queues. In this framework, it is also shown in [5] that finding the optimal *cyclic* policy is NP-complete.

In this paper, we tackle the problem of analyzing the PoA in open queueing systems of parallel and non-observable queues. We undertake this analysis from a new point of view: In contrast with the existing works above, the key point of our analysis is to consider routers with the memory of previous dispatching choices. Dispatching schemes with memory, e.g., round-robin, can be easily implemented in network routers with very limited costs. Because of the intrinsic intractability

[1]This work is supported by the Checkbound project (ANR-06-SETI-002).

of the problem, our analysis is performed in the regime where demands, i.e., job arrival rate, proportionally grow with the network size.

First, we provide a stochastic comparison result providing a lower bound on the (mean) socially-optimal response time achievable by the system. This bound is expressed in terms of a convex optimization program that integrates the mean response time of a parallel system of independent $\Gamma/M/1$ queues. Then, we introduce the "Price of Forgetting" (PoF), defined as the ratio between the socially-optimal response time of a memoryless router with respect to its memory counterpart. We prove that the PoF is bounded from above by two, implying that the PoA achieved with a router having memory can be seen as the PoA of a Bernoulli router times a correcting factor less than two. When homogeneous queues are considered, an explicit expression is found for the PoF which equals the PoA. Here, we prove that it strictly increases in the queues utilization. In contrast with the case of memoryless routers [6], [17], thus, it is not possible to design a network where the choices of selfish jobs have no impact on performance.

Our analysis also allows us to assess the quality of heuristics for the optimal (non-Bernoulli) routing. An exhaustive numerical analysis reveals that a router forwarding jobs to queues according to a given *billiard* scheme [18] yields a response time which is remarkably close to our lower bound. In other words, we have the two-fold result that billiard routings achieve, in practice, the minimum response time which, in turn, is very-well captured by our bound and approximations.

Finally, we give numerical evidence of the fact that the PoF is insensitive to the network size (N) and heterogeneity, while monotonically depending on the network load (L) only. These structural properties entail that the PoA admits the product-form $f(N)PoF(L)$ where i) $f(N)$ is linear and refers to the PoA of a Bernoulli router (which is well-understood [6], [17]), and ii) $PoF(L)$ is increasing in L and bounded from above by two for any network size.

This paper is organized as follows. Section II introduces the model under investigation and the necessary preliminaries. In Section III, we provide an upper bound on the PoA and an improved approximation. In Section IV, we define the PoF which is analyzed to derive qualitative properties on the benefit of having a router with memory. In Section V, we show how a router with memory should operate to minimize response time, and, in Section VI, we measure its impact on system performance exhibiting new structural properties. Finally, Section VII draws the conclusions of this work. We point the reader to [3] for the proofs of our results.

II. Model and Preliminaries

We consider a queueing system composed of N infinite-room queues working in parallel. Jobs arrive from an external Poissonian source having intensity λN to a router which instantaneously dispatches jobs to one of the N queues according to a given *policy*, i.e., routing rule. We assume that the router cannot observe the state of the queues, i.e., their current number of jobs, but knows the service rates of all queues, i.e., μ_i. In queue i, we assume that jobs require service for an exponentially-distributed amount of time having mean $\mu_i^{-1} = O(1)$. The service times are i.i.d. and independent of the arrival process and N. The scheduling discipline of each queue is assumed to be First-Come-First-Served. For a router with memory, the routing policy $a \stackrel{\text{def}}{=} (a^1, \ldots, a^N)$ of jobs into queues is given by the sequences $(a_n^i)_{n\in\mathbb{N}} \in \{0,1\}$, where $a_n^i = 1$ if the n^{th} job is sent to queue i, and is 0 otherwise. By definition, if $a_n^i = 1$ then $a_n^j = 0$ for all $j \neq i$, i.e., a job is routed to a single queue. Let

$$L \stackrel{\text{def}}{=} \lambda N / \sum_{i=1}^{N} \mu_i \tag{1}$$

be the *network load*, or "network utilization".

We also denote by $R = R(a)$ the mean response time, or sojourn time, of jobs in the system under policy a, provided the expectation exists (the dependence of a will be reported when necessary). In the remainder of the paper, we omit the words "mean" when we refer to response time for simplicity.

A. Nash Equilibrium and Social Optimization Revisited

Within the model introduced above, we consider two different scenarios. In the first one, selfish jobs choose to join queues to minimize their response time individually, and we refer to this situation as *Nash equilibrium* as in [6], [17]. It is assumed that incoming jobs know their arrival rate as well as the service rates of all queues. The response time achievable in this scenario is denoted by R^{Ne} and is obtained uniquely by Wardrop's principles [22]. In the second one, jobs are sent to queues by a router that minimizes the response time of all jobs simultaneously. We refer to this situation as *social optimization*, and the response time achieved in this scenario is denoted by R^{So}. These scenarios reflect the conflicting situations where jobs are non-cooperative and move in an infrastructure with neither control nor shared information with respect to the case where a centralized object dictates the dynamics of the system to maximize the profit of all jobs simultaneously.

Within these established scenarios, the fact that selfish jobs base their decisions on the arrival and service rates only does not imply that the router must base its decisions on these parameters only: Our notion of social optimization differs from the one considered in existing approaches in the sense that *we let the router operate with the memory of its previous decisions*. This means that the set of policies handled by the router is much larger than the set of the Bernoulli ones because the routing decisions are no more independent each other. In the case of Nash equilibrium, however, we observe that the decisions of jobs must be Bernoulli because jobs make their decisions independently of the others (in fact, no shared information is available in a fully-decentralized system before the arrivals of jobs). As a consequence, existing works apply to our model in this case (see [6], [17], [26] for formulae and bounds on R^{Ne}).

It is clear that both the situations depicted above can be modeled in our queueing system by specifying a suitable policy in the router.

We measure the inefficiency of the Nash equilibrium with respect to the social optimization scenario by means of the price of anarchy (PoA), which we define as follows

$$PoA \stackrel{\text{def}}{=} R^{Ne}/R^{So} \geq 1. \tag{2}$$

Evidently, large values of PoA indicate that the impact of a centralized control drastically improves the performance of the system, and vice versa. On the other hand, it is clear that a centralized system is less scalable and reliable than a distributed one because the system has a single point of failure. In (2), it is implicit that arrival and service rates are kept fixed. Notwithstanding, our results adapt to the case where (2) is extended to take into account the sup over these parameters. This follows by a relationship between our PoA and the "price of forgetting", which is introduced in Section IV.

III. Analysis

In this section, we develop an approximation and a lower bound on the socially-optimal response time by means of convex programming. Approximations and bounds on the PoA immediately follow by (2).

A. Bounding the Optimal Response Time

We now introduce a lower bound on the socially optimum response time R^{So}. Let $q_n^i(a_1^i, \ldots, a_n^i)$ be the amount of work in queue i after n arrivals. We denote by

$$Q_n^i(a_1^i, \ldots, a_n^i) \stackrel{\text{def}}{=} \mathbb{E}q_n^i(a_1^i, \ldots, a_n^i), \tag{3}$$

the mean work where the expectation is taken over all arrival times and service times. Now, let $R_i(a^i)$ be the Cesaro limit of Q_n^i, i.e.,

$$R_i(a^i) = \lim_{m\to\infty} \frac{1}{m} \sum_{n=1}^{m} Q_n^i(a_1^i, \ldots, a_n^i). \tag{4}$$

Using the PASTA property, e.g., [7], $R_i(a_i)$ is the mean response time of the jobs sent to queue i.

For any $0 < \delta < 1$, let $p_\delta^i \stackrel{\text{def}}{=} (1-\delta)\sum_{k=1}^{\infty} \delta^{k-1} a_k^i$, that exists since all a_n^i are bounded. By definition of p_δ^i, $\sum_{i=1}^{N} p_\delta^i = 1$. Therefore, the set $\mathcal{L}$ of limit points of $(p_\delta^1, \ldots, p_\delta^N)$, when $\delta \to 1$, also has a sum equal to one.

We also define the regular sequence with rate p and phase θ: for all $n \geq 1$, $\alpha_n^i(p,\theta) \stackrel{\text{def}}{=} \lfloor np+\theta \rfloor - \lfloor (n-1)p+\theta \rfloor$. Note that $\alpha_n^i(p,\theta) \in \{0,1\}$ for all n as long as $p \leq 1$ and is periodic in θ with period 1.

Theorem 1: Under the foregoing notations, the mean response time of a job under policy a verifies

$$R(a) \geq \inf_{(p_1,\ldots,p_N)\in\mathcal{L}} \big(p_1 R_1(\alpha(p_1,0)) + \cdots + p_N R_N(\alpha(p_N,0))\big).$$

The theorem says that the average response time of any policy is larger than the combination of response times in all queues where the arrival process in each queue is a regular sequence. This result is to be compared with [15] where regular sequences with rate r are proved to be optimal admission sequences in a single queue under the constraint that a proportion of at least r jobs have to be admitted in the queue. The main difference comes from the fact that routing to several queues is more difficult than admitting to a single queue because one does not know whether the proportion of jobs sent to each queue by the optimal routing policy exists. This is still an open problem and Theorem 1 above does not answer to this question but just provides a lower bound on the response time of the optimal policy. For the proposed lower bound, such proportions exist in all queues and are equal to the rates p_i achieving the infimum. On the other hand, the result stated in Theorem 1 is very close to Theorem 25 in [2]. The main difference is the fact that our cost is not additive, making the proof slightly more involved.

Let us consider a single queue i and the arrival process induced by $\alpha(p_i,\theta)$ in queue i. Let $k \stackrel{\text{def}}{=} \lfloor 1/p_i \rfloor$. The inter-arrival process $\tau_1, \cdots, \tau_n, \cdots$ has a distribution that is made of a sequence of sums of k (or $k+1$) i.i.d. exponential distributions, with parameter $N\lambda$. For example, if $p_i = 2/7$, then $k = 3$ and the arrival process in queue i under policy $\alpha(2/7, 0) = 0,0,1,0,0,0,1,0,0,1,\cdots$ where the sequences $0,0,1$ and $0,0,0,1$ alternate, is such that the distribution of the inter-arrival times alternates between the sum of three exponentials with rate $N\lambda$ and the sum of four exponentials with rate $N\lambda$.

In general, the arrival rate in queue i is $p_i N\lambda$. Now , considering a stationary i.i.d. arrival process $T_1, \cdots T_n \cdots$, with a Gamma distribution for inter-arrival times, with parameters p_i and $N\lambda$. It should be clear that for any n, these two inter-arrival processes compare for the convex ordering of random sequences: $(\tau_1, \cdots, \tau_n) \geq_{cx} (T_1, \cdots, T_n)$.

Using the fact that the mean response time of jobs is a convex increasing function of the input process (by Lindley's formula), this implies that the mean response time in queue i under a regular arrival process with rate p_i is larger than the mean response time in queue i with Gamma-distributed inter-arrivals with rate $p_i N\lambda$. This argument and Theorem 1 yield the following result.

Corollary 1: Let $R_i^{\Gamma(a,b)/M/1}$ be the mean response time of a job in queue i having exponential service times and i.i.d. inter-arrival times with a $\Gamma(a,b)$ distribution. Then,

$$R^{So} \geq \inf_{\substack{\pi_1,\ldots,\pi_N\geq 0:\\ \pi_1+\cdots+\pi_N=1}} \sum_{i=1}^{N} \pi_i R_i^{\Gamma(1/\pi_i, N\lambda)/M/1}. \tag{5}$$

In the following, we will use a coefficient that scales with N for the proportion of jobs sent to queue i: we define $\beta_i \stackrel{\text{def}}{=} N\pi_i$, where β_i is a positive constant. A lower bound on R^{So} is finally obtained by solving the following optimization problem

$$\begin{array}{rll} GB(N) \stackrel{\text{def}}{=} & \min & \displaystyle\sum_{i=1}^{N} \frac{\beta_i}{N} R_i^{\Gamma(N/\beta_i, N\lambda)/M/1}(\beta_i) \\ & \text{s.t.} & \sum_{i=1}^{N} \beta_i = N \\ & & U_i(\beta_i) \leq 1, \ \forall i \\ & & \beta_i \geq 0, \ \forall i, \end{array} \tag{6}$$

where

$$U_i(\beta_i) = \lambda\beta_i/\mu_i \tag{7}$$

and $GB(N)$ stands for Gamma-Bound with N queues. By means of Little's law, the quantity U_i is interpreted as the *utilization* of station i, i.e., the "proportion of time" in which station i is busy (in the long term).

B. Heavy-Traffic Behavior

The proposed bound $GB(N)$ is interpreted as the response time achieved when the input arrival processes of all queues are independent and Gamma distributed. This means that the router can be now thought as Bernoulli, provided that its job arrival process is no more Poisson. Since all queues become independent $\Gamma/M/1$ queues, classic heavy-traffic analysis immediately applies to derive useful approximation and insights. This corollary follows by Theorem 1 and the heavy-traffic analysis of GI/GI/1 queues; e.g. [20], Section 2.1.

Corollary 2: As $L \to 1$, $R^{So}_{Bernoulli}(N)/GB(N) \leq 2$.

It is possible to show that the inequality above becomes tighter and tighter as N increases (see the proof in [3]). In other words, our bound is essentially half of the response time of the optimal Bernoulli routing in heavy-traffic. This reveals one important structural property that we anticipate here and extend in next sections: The added-value of a router with memory is independent of the network heterogeneity when N is large and $L \to 1$.

C. Asymptotic Analysis of the $\Gamma/M/1$ Queue

In previous section, we established a lower bound on the social optimum R^{So} in terms of an optimization problem involving the response time of parallel $\Gamma/M/1$ queues. The integration of the exact $\Gamma/M/1$ (or $G/M/1$) analysis [7] in the constraints of (6) renders a non-linear problem which seems to be difficult to analyze, e.g., in terms of convexity, and also yields numerical instabilities related to $O(N^N)$ terms. Therefore, we now address the development of simple formulae for the response time of $\Gamma/M/1$ queues. These are exact in the regime where the job arrival rate to the router proportionally grows with the number of queues, for which the *ideal* job arrival processes of each queue considered by our bound $GB(N)$ become more and more deterministic.

The following theorem provides bounds on $R_i^{\Gamma(N/\beta_i,N\lambda)/M/1}$ for any network load and size.

Theorem 2: Let $\sigma_i, {\sigma_i}^+ \in [0,1)$ be the (unique) solutions of the equations

$$z\text{exp}(\tfrac{1-z}{U_i}) = 1 \tag{8}$$

and

$$z\text{exp}(\tfrac{1-z}{U_i})\left(1 - \frac{1}{2}\frac{(1-z)^2}{NU_i^2}\right) = 1, \tag{9}$$

respectively, where U_i is given by (7). Then,

$$\frac{1}{\mu_i(1-\sigma_i)} \leq R_i^{\Gamma(N/\beta_i,N\lambda)/M/1} \leq \frac{1}{\mu_i(1-\sigma_i^+)}. \tag{10}$$

Given that, as $N \to \infty$, $\sigma^+ \to \sigma$, the following corollary is straightforward.

Corollary 3: As $N \to \infty$,

$$R_i^{\Gamma(N/\beta_i,N\lambda)/M/1}(\beta_i) \to \frac{1}{\mu_i(1-\sigma_i)} \tag{11}$$

from above.

Lemma 1: $\sigma_i \leq U_i^2$.

Rewriting (8) as

$$-\frac{z}{U_i}e^{-\frac{z}{U_i}} = -\frac{1}{U_i}e^{-\frac{1}{U_i}} \tag{12}$$

and observing that it admits exactly two positive roots when $0 \leq U_i \leq 1$, where the largest one is at $z = 1$, we note that σ_i can be expressed in terms of the Lambert W function [11] if and only if $-z/U_i \geq -1 = -W(-1/e)$, which is true by Lemma 1. Hence,

$$\sigma_i = -U_i W(-\frac{1}{U_i}e^{-\frac{1}{U_i}}). \tag{13}$$

where W is the principal Lambert function. We recall that the Lambert W function [11], defined as the inverse function of $f(W) = W\text{exp}(W)$, over $[-1, +\infty)$, satisfies $0 \leq -W\left(-\frac{1}{U_i}e^{-\frac{1}{U_i}}\right) \leq 1$ for all $0 \leq U_i \leq 1$. In particular, $-W(-1/e) = 1$ and $W(0) = 0$.

The rate of convergence of Formula (11) is strictly related to the convergence of $(1 + a/N)^N$, for a fixed, to its limiting value $\text{exp}(a)$, which is known to be $\Theta(1/N)$. We will numerically show that this suffices to obtain very accurate response time estimates even when N is relatively small and it provides improved accuracy with respect to heavy-traffic approximations.

D. Approximations for Large Network Sizes

The simplicity of Formula (11) allows for the development of a simple optimization procedure. In fact, problem (6) can be rewritten as follows

$$\begin{aligned} GB(\infty) \stackrel{\text{def}}{=} \quad & \min \quad \frac{1}{\lambda N}\sum_{i=1}^{N}\frac{U_i}{1-\sigma_i(U_i)} \\ & \text{s.t.} \quad \textstyle\sum_{i=1}^{N}\frac{\mu_i}{\lambda}U_i = N \\ & \quad\quad 0 \leq U_i \leq 1, \ \forall i, \end{aligned} \tag{14}$$

where $\sigma_i(U_i)$ is given by (13), which is exact as $N \to \infty$.

Remark 1: For any N, $GB(\infty) \leq GB(N) \leq R^{So}(N)$.

Let also $GB_N(\infty)$ be the optimum of (14) where $\sigma_i(U_i)$ is given by (9). Even though $GB_N(\infty)$, in general, does not seem to provide upper bounds on the optimal response time, the following result ensures that it always provides improved accuracy with respect to $GB(\infty)$ when estimating R^{So}.

Theorem 3: $R^{So} - GB(\infty) > |R^{So} - GB_N(\infty)|$.

Even though more accurate approximations than $GB(\infty)$ and $GB_N(\infty)$ for R^{So} can be derived (by taking into account more expansions terms, see the proof of Theorem 3 [3]), we will numerically show that they suffice to obtain very accurate results. The following result ensures that efficient algorithms can be applied to solve (14) in polynomial time [8].

Theorem 4: The optimization problem (14) is convex.

The proposed upper bound on the PoA (2) follows by taking the ratio between R^{Ne} [6] and $GB(\infty)$. A numerical evaluation of its tightness and convergence speed is postponed in the experimental results section.

IV. Price of Forgetting

We define the "Price of Forgetting" (PoF) as the ratio between the socially-optimal response time achieved with a Bernoulli router and a router with memory, i.e.,

$$PoF \stackrel{\text{def}}{=} R^{So}_{Bernoulli}/R^{So}. \tag{15}$$

The following connection is immediate.

Proposition 1: The PoA (2) is given by the product between the PoF (15) and the PoA of a memoryless router.

Since the PoA in the Bernoulli case is well-understood, we limit the focus on the PoF.

A. Heterogeneous Queues

The behavior of the PoF in heavy-load follows by the discussion in Section III-B. Coincisely, $\lim_{L\to 1} PoF(L) \leq 2$ for any network size. In contrast, $\lim_{L\to 0} PoF(L) = 1$ in light-load conditions, which is intuitive: If the queue lengths are almost empty, then the response time approaches the service time of the fastest queue in any case.

The following theorem extends the heavy-traffic limit above.

Theorem 5: For any network load and size, $PoF \leq 2$.

We will also numerically show that PoF is increasing in L.

B. Homogeneous Queues

A scenario of practical interest is the case where the queues are homogeneous, i.e., $\mu_i = \mu, \forall i$, for which we can draw additional results and easily compare with Bernoulli routers.

Remark 2: If the router has no memory, then the socially optimum response time coincides with the response time in Nash equilibrium and they both admit a very simple formula:

$$R^{So}_{Bernoulli} = R^{Ne} = \frac{1}{\mu(1-L)}, \tag{16}$$

where $L = U = \lambda/\mu$ (see [17], [12]).

In other words, the PoA, in the context of memoryless routers, becomes one regardless of the utilizations.

Remark 3: If the router has memory, then (16) implies that the PoF equals the PoA (when the queues are homogeneous).

The following result is known in the literature (and also follows from Theorem 1 using a symmetry argument); see, e.g., [25] Prop. 8.3.4.

Theorem 6 ([25]): Under the foregoing assumptions, the round-robin (or cyclic) policy minimizes the mean response time for any N.

The results which follow in the remainder of this section are implicitly assumed to hold in the considered limiting regime, i.e., when $N \to \infty$ (thus providing bounds for finite N).

The following result is an immediate consequence of Theorems 2, 6 and Formula (16), and provides an asymptotically-exact formula for the PoF.

Corollary 4:

$$PoF(L) = PoA(L) = \frac{1 + LW(g(L))}{1-L} \tag{17}$$

where $g(L) = -\exp(-1/L)/L$ and $L = U = \lambda/\mu$.

As first consequence, the PoA now becomes a function of the utilization U (note that L boils down to U here), which is in contrast with the case of memoryless routers. In other words, *it is not possible to design a network where the behavior of selfish jobs has no impact on response time as in the memoryless case*. Formula (17) is thus interpreted as the correcting factor that should be taken into account by a Bernoulli analysis of the PoA. Secondly, the expression (17) lets us derive more results than Theorem 5.

Corollary 5: $PoA(L)$ is strictly increasing in L and

$$\lim_{L\to 0} \frac{\mathrm{d}PoA(L)}{\mathrm{d}L} = 1, \quad \lim_{L\to 1} \frac{\mathrm{d}PoA(L)}{\mathrm{d}L} = 0. \tag{18}$$

The limits in (18) and the monotonicity of $PoA(L)$ show that i) the response-time benefits of a router with memory are non-negligible even when the utilizations are small, and that ii) $PoA(L)$ is concave in heavy-load conditions (concavity does not seem to hold for $PoA(L)$ in general), meaning that large improvements can be obtained even in a non-negligible neighborhood of $L = 1$.

V. Optimal Routing

The framework introduced above allows us to numerically inspect the response-time gap between our lower bound and approximation and heuristic policies for the optimal routing, yielding upper bounds.

In this section, we first perform a validation of Formula (11) on several models. Then, we measure the performance achieved with a router assigning jobs to queues according to a given billiard sequences, e.g., [18], [15]. We show that the resulting distance from our formulas is remarkably small. Coincisely, our conclusions are that i) *the billiard routing scheme minimizes the response time*, and ii) *our bound and approximation on the minimum response time are tight*.

A. Accuracy of Formula (11)

We now measure the accuracy of asymptotic formula (11) by means of the percentage relative error

$$100\%|R^{\Gamma/M/1}_{exact} - R^{\Gamma/M/1}_{approx}|/R^{\Gamma/M/1}_{exact}, \tag{19}$$

where $R^{\Gamma/M/1}_{exact}$ is obtained numerically through the (exact) standard analysis of the $G/M/1$ queue, and $R^{\Gamma/M/1}_{approx}$ is given by (11). We initially evaluate (19) by varying $N \in \{50, 100, 200, 1000\}$ and $U \in \{0.1, 0.2, \ldots, 0.9, 0.95\}$. Since the mean arrival rate λ affects (19) only through the utilization, it is not considered in our experiments. Figure 1 illustrates the quality of (19) over these cases. As N grows, the error (19) decreases quickly. For $N = 50$, (11) is remarkably accurate and yields a relative error always less than 2%.

B. Quasi-Optimality of Billiard Sequences

We consider the case where the router forwards jobs to queues according to billiard sequences, which are constructed through the SG algorithm introduced in [18] (easily implementable in network routers with a very limited cost). We recall that a billiard sequence is given by the sequence of facets

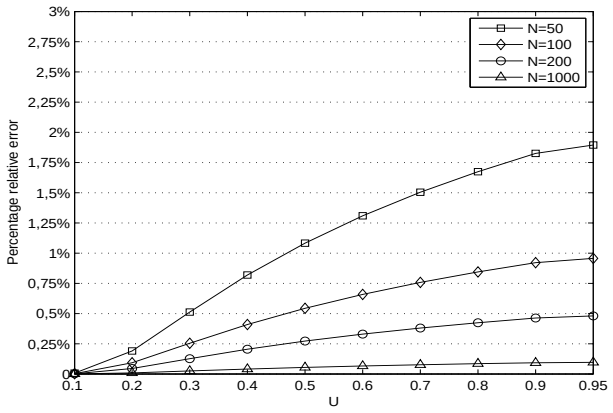

Figure 1. Accuracy evaluation of the asymptotic formula (11) through the error measure (19).

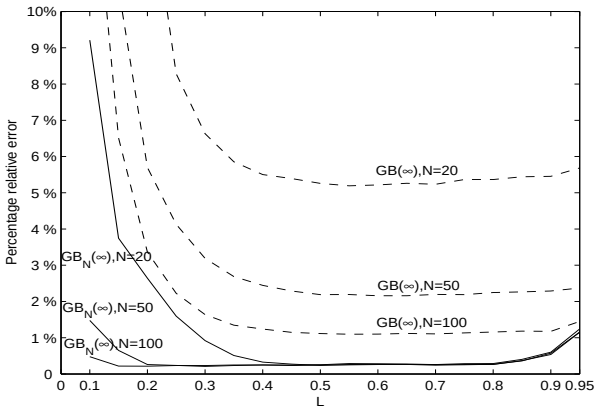

Figure 2. Plots of the error (20) averaged over a large number of tests.

(mapped to the N queues) hit by a ball in a N-dimensional cube. The SG algorithm takes as input the fraction of jobs to send to the queues (given by the solution of (14)) and an initial-position vector $x \in \mathbb{R}^N$ which we assume such that $x_i = 1$ if $\mu_i = \max_j \mu_j$ and 0 otherwise (we point the reader to [18] for further details on the SG algorithm and billiard sequences). Given that a numerical solution of the response time induced by billiard sequences is impractical for a number of reasons, e.g., the aperiodicity of the resulting routing patterns, we use simulation. To measure the gap between the response time achieved with this routing scheme and our bounds/approximations, we assess the general quality of the percentage relative error

$$\text{Err}_{\text{App}} = 100\%|R_{\text{App}} - R_{\text{Sim}}|/GB_N(\infty) \qquad (20)$$

where $R_{\text{App}} \in \{GB(\infty), GB_N(\infty)\}$ (defined in Section III-D) and R_{Sim} is the average response time computed by simulation. We measure percentage relative errors with respect to $GB_N(\infty)$ because it represents the closest approximation of R^{Opt} (see Theorem 3). The measures of R_{Sim} refer to 99% confidence intervals having size no larger than 1% of R_{Sim} itself. For any pair (N, L), $N \in \{20, 50, 100\}$ and $L \in \{0.10, 0.15, 0.20, \ldots, 0.95\}$, we generated 1,000 random models where the service rates μ_i have been drawn in the range $[0.01, 100]$ according to a uniform distribution. Larger values of N have not been considered because of the strong computational requirements of simulation. However, the proposed analysis sufficies to assess the accuracy of our approach.

The experimental results of this analysis are summarized in Figures 2, which refers to a total of nearly 50,000 experiments. In the figure, the dashed (continuous) lines refer to the error obtained with $GB(\infty)$ ($GB_N(\infty)$) for different network sizes. We clearly see that the response time achieved through a billiard routing is remarkably close to our approximation $GB_N(\infty)$ and also to our bound $GB(\infty)$. Given that the optimal response time achievable by the system must lie between our bound and the response time achieved by the billiard routing, we conclude, in an empirical sense, that billiard sequences are optimal for the response time which is in turn very-well approximated by our analysis.

The computational requirements for the computation of (14) can be found in [3], where we show that models with thousands of queues are solved in a few seconds.

VI. The Impact of Routers with Memory: Structural Properties

We now measure the proposed upper bound on the PoF in order to numerically investigate its fundamental properties. Following the results of previous section, we remark that it is very tight. Here, we infer an important structural property: *the PoF only depends on the network load L, meaning that it is independent of the network heterogeneity and size.*

A. Homogeneous Queues

In the case of homogeneous queues, the proposed bound boils down to the simple formula (17), which is asymptotically exact. By varying the utilization from 0.05 to 0.95 with step 0.05, Figure 3 illustrates i) the asymptotic PoA (17) (the dashed bold line), ii) the PoA obtained with a memoryless router (the dashed-dotted line), and iii) for $N \in \{10, 100, 1000\}$, the exact PoA, which is obtained by applied standard analysis of the $E_N/M/1$ queue. In that figure, we first notice that the PoA is not concave and (slightly) increases as N does converging to our asymptotic formula (17). The fact that the PoA increases with N finds the simple intuition that adding new resources gives more and more freedom to the router for optimizing the response time with respect to its Bernoulli counterpart. On the other hand, the PoA in the case of memoryless router remains constant to one for any number of queues and utilizations, and it is remarkably far from the ones where the router has memory. The exact PoA computed for $N = 100$ is very close to our asymptotic formula and, for $N = 10$, it has almost the same behavior. When $N = 100$ and $U = 0.85$, Figure 3 shows that a Bernoulli-based analysis underestimates the price of anarchy of a factor 1.9. When the utilizations are 0.1, i.e., small, the Bernoulli PoA is 10% lower. These observations quantify how large can be the worst-case impact of considering routers with memory in the design of

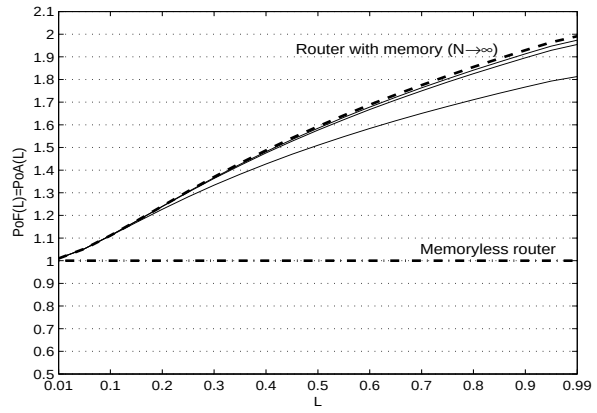

Figure 3. Price of anarchy (17) (equivalent to price of forgetting here) by varying the queues utilization. The three continuous lines correspond to the exact prices of anarchy for increasing network sizes, where the lowest (largest) refers to $N = 10$ ($N = 1000$).

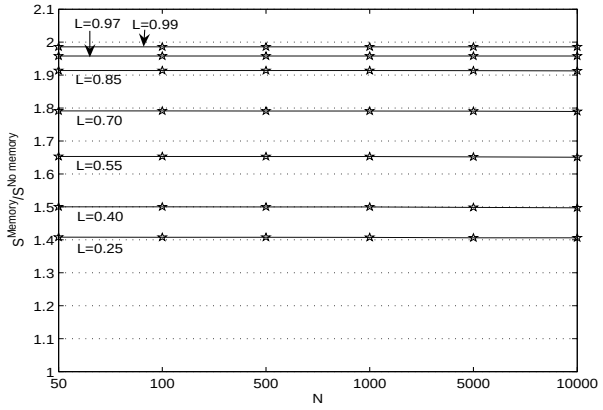

Figure 4. Insensitivity of the price of forgetting with respect to network size.

distributed or centralized systems, where utilizations usually range in $[0.6, 0.85]$.

B. Heterogeneous Queues: Independence of Network Heterogeneity and Size

We now measure the PoF in the heterogeneous case. We first consider an illustrative example which we use to inspect fundamental properties. Then, we carry out an extensive numerical analysis to give evidence of their correctness.

a) An illustrative scenario: We consider a clustered network composed of N queues where $1/10$ of the queues have fast service rates $\mu_f = 100$, $2/10$ of the queues have medium service rates $\mu_m = 50$, and the remaining ones have low service rates $\mu_l = 1$. By varying the network load (L) and size (N), we plot the resulting PoF in Figure 4, which lets us draw two important hypotheses.

First, we observe that *our bound on the PoF is independent of the network size*.

Second, if the ratios of Figure 4 are compared pointwisely to the corresponding ones of Figure 3 (where the concepts of network load and utilization are equivalent) we note that these points are very close each other. This suggests that *our bound on the PoF is not influenced by the heterogeneity of the considered scenario*, as L varies, becoming a function of the network load only. In Section III-B, we showed that this property holds true in heavy-traffic and as $N \to \infty$.

b) Exhaustive numerical investigation: We now carry out an extensive numerical analysis to give evidence of the independence of the PoF on the network size and heterogeneity. To do this, we focus on a very large test-bed of randomly generated models drawing the service rates μ_i in $[0.01, 100]$ uniformly. For any pair (N, L), we generated 1,000 models computing average and standard deviation of the PoF. The results of this analysis are shown in Table I, which refers to a total of 48,000 different models. The results presented in that

Averages

N	50	100	500	1,000	5,000	10,000
$L = 0.10$	1.252	1.254	1.254	1.254	1.253	1.253
$L = 0.25$	1.408	1.409	1.409	1.409	1.409	1.409
$L = 0.40$	1.534	1.534	1.535	1.534	1.535	1.534
$L = 0.55$	1.652	1.652	1.652	1.652	1.652	1.652
$L = 0.70$	1.768	1.768	1.768	1.768	1.768	1.768
$L = 0.85$	1.885	1.885	1.885	1.885	1.885	1.885
$L = 0.95$	1.966	1.966	1.966	1.966	1.966	1.966
$L = 0.99$	1.992	1.992	1.992	1.992	1.992	1.992

Standard deviations

N	50	100	500	1,000	5,000	10,000
L =0.10	3.0e-2	2.0e-2	8.3e-3	7.9e-3	6.9e-3	6.1e-3
L =0.25	1.4e-2	1.0e-2	5.3e-3	4.8e-3	2.8e-3	1.8e-3
L =0.40	9.5e-3	6.9e-3	3.1e-3	2.3e-3	2.1e-3	8.9e-4
L =0.55	5.3e-3	3.9e-3	1.7e-3	1.3e-3	7.1e-4	6.4e-4
L =0.70	2.9e-3	2.1e-3	1.0e-3	8.3e-4	5.5e-4	5.3e-4
L =0.85	2.1e-3	1.6e-3	7.7e-4	6.4e-4	4.8e-4	4.5e-4
L =0.95	7.3e-4	5.9e-4	4.7e-4	4.5e-4	4.4e-4	4.3e-4
L =0.99	4.8e-4	4.6e-4	4.3e-4	4.4e-4	4.3e-4	4.2e-4

Table I
AVERAGES AND STANDARD DEVIATIONS OF OUR BOUND ON THE PRICE OF FORGETTING OVER THE LARGE NUMBER OF TESTS (E-N READS 10^{-n}).

table robustly confirm the two hypotheses arisen in previous section. When $N = 50$, we observe that the averages of the PoF are already settled to their asymptotic value. Furthermore, the standard deviations are very small and decreasing in both N and L. This shows the independence with respect to the network heterogeneity. By varying L and U (for $L = U$), Figure 5 plots (17) and the average PoF shown in Table I to stress independence with respect to heterogeneity. Both curves are remarkably close each other, and they are almost equivalent when $L \geq 0.55$. In the figure, we observe that the slight gap achieved when L is small must go to zero as $L \to 0$ because, in this regime, the optimal Bernoulli and non-Bernoulli response times equal the service time of the fastest queue.

Except for the heavy-traffic case in Section III, this is surprising because *the optimal fractions of jobs sent to each queue in the Bernoulli and non-Bernoulli settings are different* (see next section). These structural properties show that:

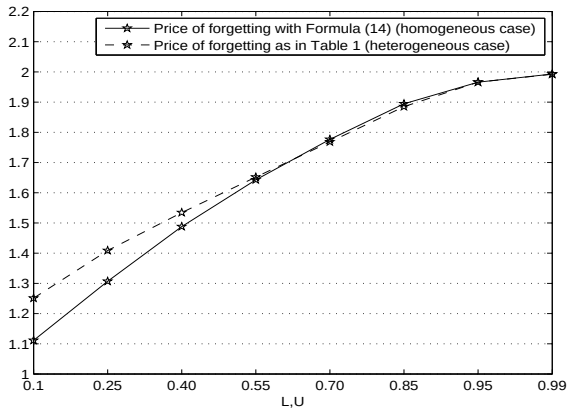

Figure 5. Comparison of Formula (17) with the averages of the prices of forgetting in Table I.

- our bound on the PoA can be seen as the product between the PoA with a memoryless router and (17). Equivalently,
- our bound (14) on the optimal response time can be seen as the ratio between the optimal Bernoulli response time and $PoF(L)$ given by (17).

The tightness of our GB bounds provides the following approximation for the optimal response time:

$$R^{So} = R^{So}_{Bernoulli}/PoF(L) \tag{21}$$

with $PoF(L)$ given by (17).

C. Optimal Routing Probabilities Comparison

We show the relation between the routing probabilities of the optimal Bernoulli router (p_i) and of our bound (14) (π_i) by evaluating the quantity $\sum_{i=1}^{N} |\pi_i - p_i|$ over the experiments performed in previous section. While in heavy-traffic the fractions of jobs in a memory/non-memory setting are equal (which is obvious) and the properties above could find some interpretation, this does not hold for the non-heavy-traffic case, for which a significant difference exists (see Table II). Notwithstanding, the PoF is not affected by such difference as shown in previous section.

L	0.25	0.40	0.55	0.70	0.85	0.95	0.99
	1.9e-1	7.7e-2	2.6e-2	6.2e-3	5.4e-4	1.3e-5	$\leq$e-5

Table II
$\sum_{i=1}^{N} |\pi_i - p_i|$ BY VARYING THE NETWORK LOAD (E-N READS 10^{-n}).

VII. Concluding Remarks

We presented a new framework for assessing the performance benefits of large centralized infrastructures with respect to their decentralized counterparts through the price of anarchy. Our analysis lets the central router exploit local information on its past routing decisions to achieve the social optimum. We showed that our revisited price of anarchy can be interpreted as the product between the corresponding memoryless price of anarchy and an increasing function of the network load only. This function approaches two in heavy-traffic, which is exact, and represents the added-value of having memory in the router. Also, we used our framework to compare routing policies for the optimal response time, numerically showing that the response time achieved by billiard sequences, which provides an upper bound, matches our lower bound. We leave as future work the case with general service time distributions.

References

[1] E. Altman, U. Ayesta, and B. Prabhu. Optimal load balancing in processor sharing systems. In *Telecomm. Syst. (to appear)*, 2010.

[2] E. Altman, B. Gaujal, and A. Hordijk. *Discrete-Event Control of Stochastic Networks: Multimodularity and Regularity*. Number 1829 in LNM. Springer-Verlag, 2003.

[3] J. Anselmi and B. Gaujal. On the price of anarchy and the optimal routing of parallel non-observable queues. Technical report, INRIA, 2010.

[4] U. Ayesta, O. Brun, and B. Prabhu. Price of anarchy in non-cooperative load balancing. Technical report, INRIA, 2009.

[5] A. Bar-Noy, R. Bhatia, J. S. Naor, and B. Schieber. Minimizing service and operation costs of periodic scheduling. *Math. Oper. Res.*, 27(3):518–544, 2002.

[6] C. H. Bell and S. Stidham. Individual versus social optimization in the allocation of customers to alternative servers. *Management Science*, pages 29:83–839, 1983.

[7] U. N. Bhat. *An Introduction to Queueing Theory: Modeling and Analysis in Applications*. Birkhauser Verlag, 2008.

[8] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, March 2004.

[9] H.-L. Chen, J. R. Marden, and A. Wierman. The effect of local scheduling in load balancing designs. *SIGMETRICS Perf. Eval. Rev.*, 36(2):110–112, 2008.

[10] M. B. Combé and O. J. Boxma. Optimization of static traffic allocation policies. *Theor. Comput. Sci.*, 125(1):17–43, 1994.

[11] R. M. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jerey, and D. E. Knuth. On the lambert w function. *Adv. Comput. Math*, pages 329–359, 1996.

[12] S. C. Dafermos and F. T. Sparrow. The traffic assignment problem for a general network. *J. Res. Nat. Bureau Standards,B*, 73(2):91–118, 1969.

[13] E. J. Friedman. Genericity and congestion control in selfish routing. In *43rd IEEE Conf. on Decision and Control*, pages 4667–4672, 2003.

[14] B. Gaujal, E. Hyon, and A. Jean-Marie. Optimal routing in two parallel queues with exponential service times. *Discrete Event Dynamic Systems*, 16(1):71–107, 2006.

[15] B. Hajek. The proof of a folk theorem on queueing delay with applications to routing in networks. *J. ACM*, 30:834–851, 1983.

[16] B. Hajek. Extremal splitting of point processes. *Math. Oper. Res.*, 10:543–556, 1986.

[17] M. Haviv and T. Roughgarden. The price of anarchy in an exponential multi-server. *Op. Res. Lett.*, 35(4):421–426, 2007.

[18] A. Hordijk and D. van der Laan. Periodic routing to parallel queues and billiard sequences. *Mathematical Methods of Operations Research*, 59(2):173–192, 2004.

[19] F. Kelly. Network routing. *Philosophical Transactions of the Royal Society A337*, pages 343–367, 1991.

[20] L. Kleinrock. *Queueing Systems, Volume 2: Computer Applications*. Wiley, 1976.

[21] E. Koutsoupias and C. Papadimitriou. Worst-case equilibria. In *STACS*, volume 1563 of *LNCS*, pages 404–413, January 1999.

[22] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, New York, NY, USA, 2007.

[23] T. Roughgarden. The price of anarchy is independent of the network topology. In *J. of Computer and System Sciences*, pages 428–437, 2002.

[24] R. Subrata and A. Y. Zomaya. Game-theoretic approach for load balancing in computational grids. *IEEE Trans. Parallel Distrib. Syst.*, 19(1):66–76, 2008.

[25] J. Walrand. *An Introduction to Queueing Networks*. Prentice-Hall, 1988.

[26] T. Wu and D. Starobinski. On the price of anarchy in unbounded delay networks. In *GameNets*, page 13, New York, NY, USA, 2006. ACM.