Dynamic control of the join-queue lengths in saturated fork-join queues

Andrea Marin and Sabina Rossi

DAIS - Università Ca' Foscari, Venezia, Italy
{marin,srossi}@dais.unive.it

Abstract. The analysis of fork-join queueing systems has played an important role for the performance evaluation of distributed systems where parallel computations associated with the same job are carried out and the job is considered served only when all the parallel tasks it consists of are served and then joined. The fork&join nodes that we consider consist of $K \geq 2$ parallel servers each of which is equipped with two FCFS queues, namely the service-queue and the join-queue. The latter store the serviced tasks waiting for being joined. This paper addresses the problem that under independent and exponentially distributed service time, the process describing the join-queue lengths becomes instable under heavy load. This is due to the variance of the service time distribution. We propose a simple mechanism that avoids this problem, show that we can analytically study a set of relevant performance indices and study by simulation its robustness.

1 Introduction

Fork-join queueing stations have been widely studied in the literature because of their wide applications in the context of distributed and parallel systems. Such queueing stations behave as follows: jobs arrive according to an arrival process and are forked into K tasks that are enqueued in the *service-queue* and then served by independent servers. Once a task is serviced, it is enqueued in the *join-queue* waiting for the service completions of all the other tasks of the job it belongs to. Once all the tasks of a job are serviced, the *join* operation is performed and the job leaves the systems.

Fork-join queues have found applications in a wide variety of domains in computer science and telecommunication networks. For instance, in [**?**] the authors study the response times of multiprocessor systems by means of fork-join networks, in [**?**] the authors consider parallel communication systems and in [**?**] a RAID system is studied by simulating a fork-join station.

In this work we introduce a simple mechanism that allows the system to dynamically control the length of join-queues by slowing down the processors that have already served many tasks while maintaining the other ones at their full speed. As a consequence, although we observe a reduction of the system's throughput, the length of the join-queues will be highly reduced and the system significantly reduces its energy consumption. Informally, the idea is that it is not

worth using a processor at its full speed when the worked task will have to wait for the join operation in the join-queue. Our contribution includes an analytically tractable model of such a rate control mechanism. We start by considering the FlattoHahnWright (FHW) model [**?**,**?**] in saturation, i.e., the service times are modelled by i.i.d. exponential random variables, the join operation is instantaneous, and the service queues are never empty. We show that even in the case of two servers ($K = 2$), the stochastic process modelling the join-queue lengths is unstable because of the variance in the service times. By the introduction of our rate-control mechanism we show that the model process underlying the join-queue length becomes stable and their expected length is finite. Moreover, we are able to derive an analytical expression for the system's throughput. We study, by simulation, the behaviour of our algorithm when the service times are not exponentially distributed and show a coefficient of variation greater or lower than one.

1.1 Related work

[**?**]

2 Rate-control algorithm

In this section we formally introduce the problem we are studying and the rate-control algorithm that we propose. In the following sections we study the performance of such an algorithm in terms of throughput, load-balance and energy saving.

2.1 Problem statement

Let us consider a fork-join queueing system with K servers as depicted in Figure 1. We consider a saturated model, i.e., there is always a job waiting for being processed. As a consequence the servers' queues always contain at least one task. The service times are modelled by i.i.d. continuous time random variables and we initially assume that the join operation occur immediately after all the tasks belonging to the same job are served. All the queues follow a FCFS discipline. Clearly, if the expected service time at the servers is not the same, and not rate-control mechanism is applied, then the join-queue length of the fastest servers tend to grow infinitely large as time $t \to \infty$. Less obvious is the case in which all the service times are independent and identically distributed, i.e., with the same mean. In these cases, the variance of the service time causes an unbounded growth of the join-queue population, i.e., the expected queue length at the servers tends to infinity as $t \to \infty$. In Figure 2 we show a transient simulation of the saturated model with three service time distributions: Erlang-2, hyperexponential and exponential. The confidence intervals have been build on 15 independent executions of the simulation with a confidence of 95%. The plot supports the intuition that higher coefficient of variations in the service times

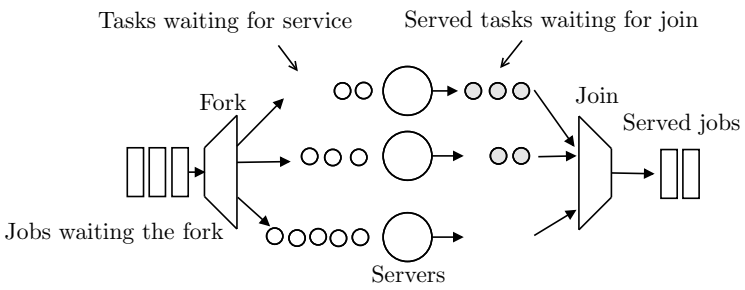


Fig. 1: Fork-join queueing station with $K = 3$ servers.

make the expected queue lengths grow faster. We formally prove the model instability for when the service times are exponentially distributed.

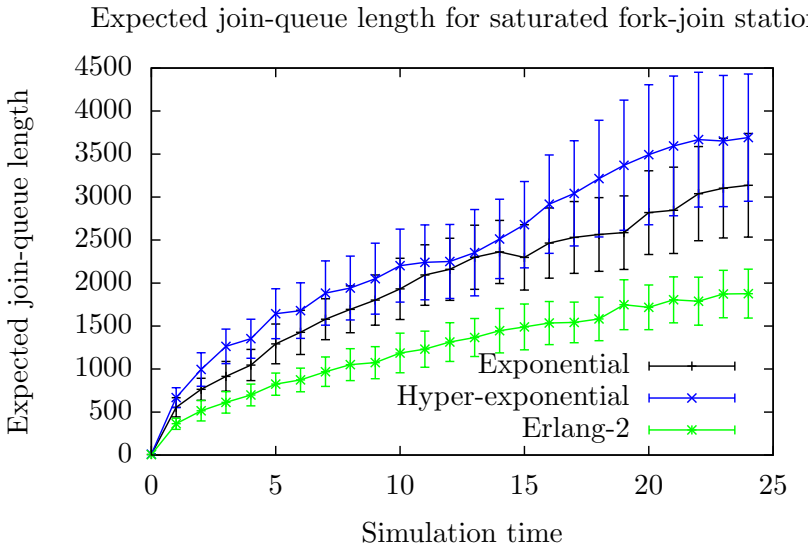


Fig. 2: Growth of the expected join-queue length for $K = 20$ servers, exponential ($CV = 1$), Erlang-2 ($CV = \sqrt{2}/2$), Hyper-exponential ($CV = 1.31$).

Proposition 1. *In the long run, the saturated fork-join model with $K \geq 2$, i.i.d. exponential service times, immediate join, has an infinite expectation of the join-queue length.*

Proof. For brevity, we give the proof for $K = 2$. The state space of the model is

$$\mathcal{S} = \{(n_1, n_2) : n_1 = 0 \vee n_2 = 0, n_i \in \mathbb{N}\},$$

and the transitions are from state $(0, n_2)$ to $(0, n_2+1)$ or to $(0, n_2-1)$ and from state $(n_1, 0)$ to $(n_1+1, 0)$ or $(n_1-1, 0)$. Since the service times are exponential the stochastic process is continuous time Markov chain, and specifically is a random walk on the line. In this CTMC all the rates are equal and hence the states are not positive recurrent. Therefore, let Q be the random variable associated with the join-queue length for one of the two servers at a time t_0, with $t_0 \to \infty$, then $E[Q] = \infty$. □

We devise an algorithm that dynamically controls the service rates (e.g., by scaling the operating frequency of the processors) with the following aims:

- Having a finite expectation of the join-queue lengths;
- Maintaining the throughput at reasonable high levels;
- Reducing the overall energy consumption by controlling the servers' rates.

Moreover, we will see that if the service rates are exponentially distributed, then a Markovian model with analytically tractable solution exists, therefore one can tackle problems of optimisation or capacity planning that would be expensive to address by stochastic simulation.

2.2 The rate-control algorithm

The main idea of the algorithm is to slow down the servers that have already completed their work on many tasks whereas the servers that have served less tasks will work at maximum speed. Since it would be unrealistic to assume that each server can take a decision about its own speed by knowing the global state of the system, we introduce a policy that implements a rate-control strategy by just maintaining a single integer state variable. Let us label each of the K servers with integer numbers in $\{1, \ldots, K\}$ and define the following neighbourhood relation: for each server k we define its neighbour $ne(k)$ as:

$$ne(k) = \begin{cases} k+1 & \text{if } k < K \\ 1 & \text{if } k = 1 \end{cases}.$$

Let n_k denote the state variable of each server. When server k completes a task, then n_k is increased by 1, when its neighbour completes a task n_k is decreased by 1. In other words, n_k maintain the difference between the join-queue length of server k and $ne(k)$. Let $\mu(n_k)$ be the local state dependent service rate at a server (recall that they are all stochastically identical), then:

$$\mu(n_k) = \begin{cases} \frac{\mu}{n_k+1} & \text{if } n_k \geq 0 \\ \mu & \text{otherwise} \end{cases}. \tag{1}$$

Intuitively, when a server k has completed less or the same number of tasks than $ne(k)$ then it works at its full service speed, otherwise it slows down in a proportional way with the number of exceeding jobs. Notice that for server k, the

key point for regulating the join-queue length is to consider the difference in the queue lengths of the servers rather that the total length of its join queue. Indeed this latter value could be high because of some delay in the join operation, while the mechanism that we propose is based on balancing the number of tasks served by each server.

3 Analytical model for the rate-control mechanism applied to the saturated FHW model with immediate join

Let us consider the vector $\mathbf{n} = (n_1, \ldots, n_K)$ of the state variables of each server, and observe that at each time epoch we have $\sum_{k=1}^{K} n_k = 0$. We aim at studying the stochastic process $\mathbf{n}(t)$ on the state space $\mathcal{S} = \{\mathbf{n} = (n_1, \ldots, n_K) : n_k \in \mathbb{Z},\ \sum_{k=1}^{K} n_k = 0\}$. Since the service rates are the only events that cause a state change, from the fact that they are exponentially distributed we conclude that $\mathbf{n}(t)$ if a homogeneous CTMC. Although we will derive a product-form expression for the invariant measure of $\mathbf{n}(t)$, it is worth of notice that $\mathbf{n}(t)$ is *not* reversible for $K > 2$. In fact, consider state $(0, 0, 0)$ and assume that server 2 completes a task taking the state of the process to $(0, 1, -1)$. It should be clear that there does not exist any transition bringing back the model to $(0, 0, 0)$. One path that brings back the model state to $(0, 0, 0)$ is that consisting of a sequence of transitions associated with one task completion at servers 1 and 3.

Before proceeding with he analysis we have to introduce the regularized Kummer's confluent hypergeometric function $\mathbf{M}(a, b, x)$ defined as follows (the first equality shows an alternative common notation):

$$\mathbf{M}(a, b, x) = {}_1\tilde{F}_1(a; b; x) = \frac{1}{\Gamma(b)} M(a, b, x) \qquad b \in \mathbb{N}^+ , \tag{2}$$

where $\mathrm{M}(a, b, x)$ is the Kummer's confluent hypergeometric function defined by the series

$$M(a, b, x) = {}_1F_1(a; b; x) = \sum_{k=0}^{\infty} \frac{(a)_k}{(b)_k} \frac{x^k}{k!} \qquad b \in \mathbb{N}^+ , \tag{3}$$

Γ is Euler's Gamma function and $(y)_k$ is the Pochhammer's symbol $(y)_k = y(y + 1) \cdots (y + k1)$.

Theorem 1. *Given the CTMC* $\mathbf{n}(t)$*, then we have that:*

1. $\mathbf{n}(t)$ *is ergodic, i.e., admits a unique stationary distribution* $\pi_K(\mathbf{n})$*;*
2. *The stationary distribution is given by the following expression:*

$$\pi_K(\mathbf{n}) = \frac{1}{G_K} \frac{1}{\prod_{i=1}^{K} (n_i \delta_{n_i > 0})!} \tag{4}$$

where we assume that empty products are equal to 1 *and* δ_P *is* 1 *if proposition* P *is true,* 0 *otherwise and*

$$G_K = 1 + \sum_{j=1}^{K-1} \binom{K}{j} j^{K-j} \mathbf{M}(K-j, K-j+1, j)\,. \tag{5}$$

We base the proof of the theorem on few Lemmas: first we assume the ergodicity and derive the model's product-form expression. The we show that normalising constant G_K is finite (thanks to the properties of the Kummer's confluent hypergeometric function) for finite K and hence the CTMC must be ergodic.

Lemma 1. *Assume that* $\mathbf{n}(t)$ *is ergodic and hence admits a unique stationary distribution. Then, its expression is that of Equation (4) where:*

$$G_K = \sum_{\mathbf{n}\in\mathcal{S}} \frac{1}{\prod_{i=1}^{K}(n_i\delta_{n_i>0})!}\,. \tag{6}$$

Proof. The proof can be obtained by substitution of Equation (4) in the system of global balance equations of the CTMC. □

Notice that since $\mathcal{S}$ is an infinite set, at the moment the fact that G_K is finite, i.e., the infinite series (6) converges, depends on the assumption of ergodicity. We now algebraically prove that (5) converges and hence that the CTMC is ergodic.

Lemma 2. *The series (6) is equivalent to the expression given by Equation (5) which is finite for any* $K \in \mathbb{N}$.

Proof. Let $\mathcal{P}(\mathbf{n})$ be the multiset with all the non-negative components of $\mathbf{n}$, i.e., $\mathcal{P}(\mathbf{n}) = \{n_i : n_i \geq 0\}$ and observe that for all the states $\mathbf{n}'$ such that $\mathcal{P}(\mathbf{n}') = \mathcal{P}(\mathbf{n})$ the expression under the sum symbol of Equation (6) is the same. Let $1 \leq j \leq K-1$ and $(x_1, \ldots, x_j)$ be a tuple such that $x_i \geq 0$ for all $i = 1, \ldots, j$ and $\sum_{i=1}^{j} x_j = n$, with $n \geq 0$. Basically, j denotes the number of non-negative components in a state and n their sum. Notice that, given j and n we can count how many states have exactly j non-negative components whose sum is n. This is given by the product of the number of non-negative solutions of the Diophantine's equation $y_1 + \ldots + y_j = n$ multiplied by the number of strictly positive solutions of the Diophantine's equation $y_1 + \ldots + y_{K-1} = n$ (since the sum of all the state components is 0), i.e., we can rewrite the normalising constant as:

$$G_K = 1 + \sum_{j=1}^{K-1} \sum_{n=K-j}^{\infty} \sum_{\mathbf{x}:x_1+\ldots+x_j=n} \frac{1}{\prod_{t=1}^{j} x_t!} \binom{K}{j} \cdot \binom{n-1}{K-j-1} = 1 + \sum_{j=1}^{K-1} \binom{K}{j} \sum_{n=K-j}^{\infty} \frac{j^n}{n!} \binom{n-1}{K-j-1},$$

where the last equality follows from the multinomial theorem. Notice that the boundaries of j in the external summatory start from 1 (there cannot be any state with all negative components) and terminate at $K-1$. Indeed, the only state with all non-negative components is $\mathbf{0}$ that we take into account by summing 1 at the beginning of the right-hand-side.
We can rewrite Equation (2) as:

$$\mathbf{M}(a,b,x) = \sum_{k=0}^{\infty} \frac{(a)_k}{\Gamma(b+k)} \frac{x^k}{k!} \qquad b \in \mathbb{N}^+. \tag{7}$$

So we have:

$$\begin{aligned}
G_K &= 1 + \sum_{j=1}^{K-1} \binom{K}{j} \sum_{w=0}^{\infty} \frac{j^{w+K-j}}{(w+K-j)!} \binom{w+K-j-1}{K-j-1} \\
&= 1 + \sum_{j=1}^{K-1} \binom{K}{j} \sum_{w=0}^{\infty} \frac{j^{w+K-j}}{(w+K-j)!} \frac{(K-j)_w}{w!} \\
&= 1 + \sum_{j=1}^{K-1} \binom{K}{j} j^{K-j} \sum_{w=0}^{\infty} \frac{j^w}{\Gamma(w+K-j+1)} \frac{(K-j)_w}{w!} \\
&= 1 + \sum_{j=1}^{K-1} \binom{K}{j} j^{K-j} \mathbf{M}(K-j, K-j+1, j)
\end{aligned}$$

where the last equality follows from Equation (7) with $a = K-j$, $b = K-j+1$ and $x = j$. Finally, we observe that $1 < G_K < \infty$ since its definition does not involve any infinite sum and function $\mathbf{M}$ evaluation at the specified integer parameters is always finite and non-negative. □

Proof of Theorem 1. The theorem follows straightforwardly by Lemma 1 and 2. □

In order to derive the expression for the marginal distribution of the join-queue lengths we have to consider that although the state space of each single queues ranges from $-\infty$ to $+\infty$, the joint state space is not the Cartesian product of the single state spaces. Therefore, the knowledge of G_K is not sufficient to obtain the marginal distribution. A similar situation arises when studying closed queueing networks. However, while for closed product-form queueing networks several algorithms have been proposed, e.g., [**?**,**?**,**?**], in our case we are able to express the marginal distributions in terms of (regularized) Kummer's function evaluated in point whose closed-form solution is known.

Let us consider the definition of G_K given by Equation (6), and let G_k^N be the normalising constant defined as:

$$G_k^N = \sum_{\mathbf{n} \in \mathcal{S}_k^N} \frac{1}{\prod_{i=1}^{k} (n_i \delta_{n_i>0})!},$$

where $\mathcal{S}_k^N = \{(n_1, \ldots, n_k) : \sum_{i=1}^k n_i = N\}$. Note that $G_K = G_K^0$. Then, we can write the marginal distribution as:

$$\pi_K^*(n) = \frac{1}{(n_i \delta_{n_i>0})!} \frac{G_{K-1}^{-n}}{G_K^0} \,. \tag{8}$$

The following Lemma gives the expression for G_k^N for arbitrary $k \geq 1$ and $N \in \mathbb{Z}$.

Lemma 3. *The expression for G_k^N is:*

- *If $N \geq 0$:*

$$G_k^N = \frac{(k\mu)^N}{N!} + \mu^N \sum_{j=1}^{k-1} \binom{k}{j} j^{N+k-j} \; \mathbf{M}(k-j, N+k-j+1, j) \,.$$

- *If $N < 0$ and $2 \leq k \leq N$:*

$$G_k^N = \binom{-N-1}{k-1} \mu^N + \mu^N \sum_{j=1}^{k-1} \binom{k}{j} \binom{-N-1}{k-j-1} M(-N, -N-k+j+1, j) \,.$$

- *If $N < 0$ and $k > N$:*

$$\begin{aligned} G_k^N = \mu^N & \sum_{j=1}^{k+N-1} \binom{k}{j} j^{N+k-j} \; \mathbf{M}(k-j, N+k-j+1, j) \\ & + \mu^N \sum_{j=k+N}^{K-1} \binom{k}{j} \binom{-N-1}{k-j-1} M(-N, -N-k+j+1, j) \end{aligned}$$

- *If $k = 1$:*

$$G_1^N = \begin{cases} \mu^N / N! & \text{if } N \geq 0 \\ \mu^N & \text{if } N < 0 \end{cases}$$

The following lemma gives an analytical expression for the station's throughput.

Lemma 4. *The throughput X of the model in steady-state is:*

$$\begin{aligned} X = \frac{\mu}{K G_K} \Bigg(K + \sum_{j=1}^{K-1} \binom{K}{j} j \Big(& j^{K-j+1} \; \mathbf{M}(K-j, K-j+2, j) \\ & - (j-1)^{K-j+1} \; \mathbf{M}(K-j, K-j+2, j-1) \\ & + (K-j) j^{K-j-1} \mathbf{M}(K-j, K-j+1, j) \Big) \Bigg) . \end{aligned} \tag{9}$$

The numerical evaluations of both G_k^N and of T_K are based on the computation of the confluent hypergeometric function $\mathbf{M}(a, b, z)$ with parameters $a \in \mathbb{N}^+$, $b \in \mathbb{N}^+$ and $b > a$. Indeed, if a and b are non-negative integers, then the series converges for all finite x. In particular, for $b > a$, $\mathbf{M}(a, b, z)$ converges to [**?**]:

$$\mathbf{M}(a,b,z) = \left(e^z \sum_{k=0}^{a-1} \frac{(1-a)_k \ (-z)^k}{k! \ (2-b)_k} - \sum_{k=0}^{b-a-1} \frac{(1-b+a)_k \ z^k}{k! \ (2-b)_k} \right) \frac{(2-b)_{a-1} \ z^{1-b}}{(a-1)!} . \quad (10)$$

4 Numerical evaluation

In this section we study the sensitivity of the throughput, the expected join-queue length and the power consumption with respect to the distribution of the service times and on the assumption of a saturated model. Then, we study the performances in terms of throughput and energy consumption of the model implementing the rate-control algorithm under the assumptions introduced in Section 3.

4.1 The power consumption

Since our rete-control mechanism reduces the computation speed of the severs, this can be interpreted as a reduction of the operating frequency leading to a reduction of the overall server power consumption. Clearly the minimum power consumption with maximum throughput corresponds to a situation in which the servers work at a constant maximum rate, but we have already discussed that the drawback of this approach is the infinite growth of the join-queue length in saturated models.

Under the assumptions of Section 3 we know the analytical expression of the marginal stationary state distribution for each server (see Equation (8) and Lemma 3). This allows us to define a lower and upper bound of the energy consumption by truncation of the probabilities. Given an integer $E > 0$, the expected power consumption in steady-state $\overline{P}$ is bounded by:

$$\sum_{i=-E}^{-1} \pi_K^*(i) + \sum_{i=0}^{E-1} \pi_K^*(i) \frac{1}{(i+1)^3} < \overline{P} < \sum_{i=-E}^{-1} \pi_K^*(i) + \sum_{i=0}^{E-1} \pi_K^* \frac{1}{(i+1)^3} + (1 - \sum_{i=-E}^{E-1} \pi_K^*) ,$$

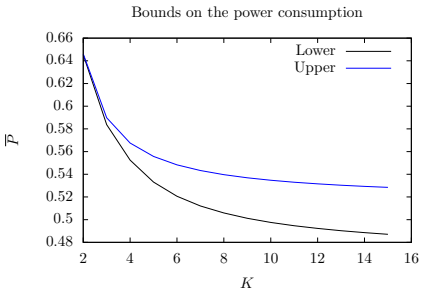
where we have assumed that the sever at maximum speed consumes 1 unit of energy for unit of time, and that the power consumption depends on the cube of the operating frequency. Clearly, more accurate models of the relation between operating frequency and power consumption can be considered, but this is out of the scope of this paper, especially because this relation depends on the intrinsic characteristics of the processors [**?**].

4.2 Sensitivity analysis

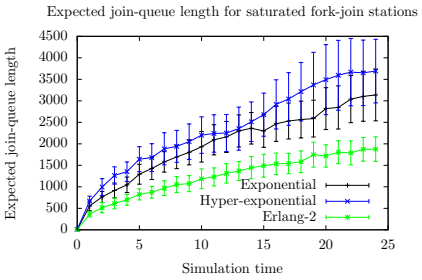
The analytical model proposed in Section 3 requires that the service time are state dependent i.i.d. exponential random variables. Under this assumption and by considering a saturated model with immediate join, we proved the stability of the process modelling the join-queue lengths. Clearly, we expect to find a sensitivity of the performance indices on the distribution of the service times, because its its variance that causes the growth the join-queue length in the model without the rate-control mechanism. It is nice to note that with small values of $E \simeq 10$ we obtain tight bounds for the energy consumption as shown in Figure 3-(A).

References

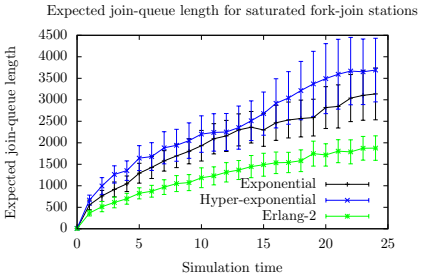
1. Optimal job splitting in parallel processor sharing queues. *Stochastic models*, 28:144–166, 2012.
2. S. C. Bruell, G. Balbo, and P. V. Afshari. Mean Value Analysis of mixed, multiple class BCMP networks with load dependent service stations. *Perf. Eval.*, 4:241–260, 1984.
3. J. P. Buzen. Computational algorithms for closed queueing networks with exponential servers. *Commun. ACM*, 16(9):527–531, 1973.
4. G. Casale. A generalized method of moments for closed queueing networks. *Perform. Eval.*, 68(2):180–200, 2011.
5. L. Flatto and S. Hahn. Two parallel queues created by arrivals with two demands. *SIAM J. on Applied Mathematics*, 44(5):1041, 1984.
6. A. S. Lebrecht, N. J. Dingle, and W. J. Knottenbelt. Modelling zoned RAID systems using fork-join queueing simulation. In *Proc. of 6th European Performance Engineering Workshop, EPEW 2009 London, UK, July 9-10, 2009 Proceedings*, pages 16–29. Springer, 2009.
7. V. Nguyen. Processing networks with parallel and sequential tasks: heavy traffic analysis and Browinian limits. *Annals of Applied Probability*, 3(1):28–55, 1993.
8. F. W. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark. *NIST Handbook of Mathematical Functions*. Cambridge University Press, New York, NY, USA, 1st edition, 2010.
9. T. Rauber and G. Rünger. Energy-aware execution of fork-join-based task parallelism. In *Proc. of the 20th IEEE International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems, (MASCOTS)*, pages 231–240, 2012.
10. D. Towsley, G. Romel, and J. Astankovic. Analysis of fork-join program response times on multiprocessors. *IEEE Trans. on Parallel and Distributed Systems*, 1(3):286–303, 1990.
11. Paul E. Wright. Two parallel processors with coupled inputs. *Advances in Applied Probability*, 24:986–1007, 1992.



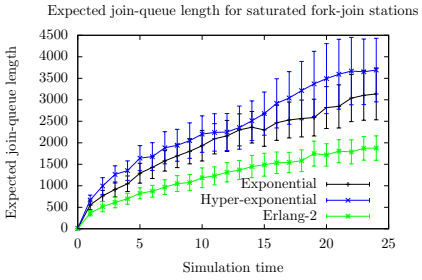
(a) Bounds of the power consumption



(b) Two



(c) Three



(d) Bounds of the expected energy consumption

Fig. 3: Numerical Evaluation.

A Proofs of statements

Proof of Lemma 3

Let us define

$$\mathcal{S}_k^N = \{\mathbf{n} = (n_1, \ldots, n_k) : n_i \in \mathbb{Z},\ \sum_{i=1}^{k} n_i = N\}\,.$$

Then the state space of the stochastic process $\mathbf{n}(t)$ is $\mathcal{S} = \mathcal{S}_K^0$.

First observe that for $N \geq 0$

$$\begin{aligned}\mathcal{S}_k^N = &\textstyle\bigcup_{j=1}^{k-1} \big(\bigcup_{p=N+k-j}^{\infty} \{\mathbf{n} = (n_1, \ldots, n_k) : n_i \in \mathbb{Z},\\ &\textstyle\sum_{i=1}^{k} \delta_{n_i \geq 0} = j,\ \sum_{i=1}^{k} n_i \delta_{n_i \geq 0} = p,\ \sum_{i=1}^{k} n_i = N\}\\ &\textstyle\bigcup \{\mathbf{n} = (n_1, \ldots, n_k) : n_i \in \mathbb{Z},\ \sum_{i=1}^{k} \delta_{n_i \geq 0} = k,\ \sum_{i=1}^{k} n_i = N\}\end{aligned}$$

Now consider $N < 0$ with $2 \leq K \leq -N$, then

$$\begin{aligned}\mathcal{S}_k^N = &\textstyle\bigcup_{j=1}^{k-1} \big(\bigcup_{p=0}^{\infty} \{\mathbf{n} = (n_1, \ldots, n_k) : n_i \in \mathbb{Z},\\ &\textstyle\sum_{i=1}^{k} \delta_{n_i \geq 0} = j,\ \sum_{i=1}^{k} n_i \delta_{n_i \geq 0} = p,\ \sum_{i=1}^{k} n_i = N\}\\ &\textstyle\bigcup \{\mathbf{n} = (n_1, \ldots, n_k) : n_i \in \mathbb{Z},\ \sum_{i=1}^{k} \delta_{n_i \geq 0} = 0,\ \sum_{i=1}^{k} n_i = N\}\end{aligned}$$

Finally we consider $N < 0$ with $K > -N$, then

$$\begin{aligned}\mathcal{S}_k^N = &\textstyle\bigcup_{j=1}^{k+N-1} \big(\bigcup_{p=N+k-j}^{\infty} \{\mathbf{n} = (n_1, \ldots, n_k) : n_i \in \mathbb{Z},\\ &\textstyle\sum_{i=1}^{k} \delta_{n_i \geq 0} = j,\ \sum_{i=1}^{K} n_i \delta_{n_i \geq 0} = p,\ \sum_{i=1}^{k} n_i = N\}\\ &\textstyle\bigcup_{j=k+N}^{k-1} \big(\bigcup_{p=0}^{\infty} \{\mathbf{n} = (n_1, \ldots, n_k) : n_i \in \mathbb{Z},\\ &\textstyle\sum_{i=1}^{k} \delta_{n_i \geq 0} = j,\ \sum_{i=1}^{k} n_i \delta_{n_i \geq 0} = p,\ \sum_{i=1}^{k} n_i = N\}\end{aligned}$$

We first compute G_k^N for $N \geq 0$. By the definition of S_k^N above, we can write

$$\begin{aligned}G_k^N &= \frac{(k\mu)^N}{N!} + \sum_{j=1}^{k-1} \sum_{p=N+k-j}^{\infty} \sum_{\mathbf{x}: x_1 + \ldots + x_j = p} \frac{\mu^p}{\prod_{t=1}^{j} x_t!} \frac{1}{\mu^{p-N}} \binom{k}{j} \binom{p-N-1}{k-j-1}\\ &= \frac{(k\mu)^N}{N!} + \mu^N \sum_{j=1}^{k-1} \binom{k}{j} \sum_{p=N+k-j}^{\infty} \frac{j^p}{p!} \binom{p-N-1}{k-j-1}\end{aligned}$$

where the last equality follows from the multinomial theorem. Notice that the boundaries of j in the external summatory start from 1 (there cannot be any state with all negative components) and terminate at $K-1$. The states with all non-negative components are considered into the addend $\sum_{\mathbf{x}: x_1 + \ldots + x_k = N} \frac{\mu^p}{\prod_{t=1}^{j} x_t!} \frac{1}{\mu^{p-N}}$

$= \frac{(K\mu)^N}{N!}$ at the beginning of the right-hand-side. The lower bound of p is $N+k-j$ following from the fact that, since $k-j$ states are negative, then p has a minimal value of $N+k-j$. Now G_k^N can be written:

$$\begin{aligned} G_k^N &= \frac{(k\mu^N}{N!} + \mu^N \sum_{j=1}^{k-1} \binom{k}{j} \sum_{w=0}^{\infty} \frac{j^{w+N+k-j}}{(w+N+k-j)!} \binom{w+k-j-1}{k-j-1} \\ &= \frac{(k\mu)^N}{N!} + \mu^N \sum_{j=1}^{k-1} \binom{k}{j} j^{N+k-j} \sum_{w=0}^{\infty} \frac{j^w}{\Gamma(w+N+k-j+1)} \frac{(k-j)_w}{w!} \\ &= \frac{(K\mu)^N}{N!} + \mu^N \sum_{j=1}^{k-1} \binom{K}{j} j^{N+k-j}\ \mathbf{M}(k-j, N+k-j+1, j) \end{aligned}$$

Now we compute G_k^N for $N < 0$ and $2 < k \leq -N$. By the definition of S_k^N above, we can write

$$\begin{aligned} G_k^N &= \binom{-N-1}{k-1}\mu^N + \sum_{j=1}^{k-1}\sum_{p=0}^{\infty} \sum_{\mathbf{x}: x_1+\ldots+x_j=p} \frac{\mu^p}{\prod_{t=1}^{j} x_t!} \frac{1}{\mu^{p-N}} \binom{k}{j} \binom{p-N-1}{k-j-1} \\ &= \binom{-N-1}{k-1}\mu^N + \mu^N \sum_{j=1}^{k-1} \binom{k}{j} \sum_{p=0}^{\infty} \frac{j^p}{p!} \binom{p-N-1}{k-j-1} \\ &= \binom{-N-1}{k-1}\mu^N + \mu^N \sum_{j=1}^{K-1} \binom{k}{j} \sum_{p=0}^{\infty} \frac{j^p}{p!} \frac{(p-N-1)!}{(k-j-1)!(p-N-k+j)!} \\ &= \binom{-N-1}{k-1}\mu^N + \mu^N \sum_{j=1}^{k-1} \binom{K}{j} \binom{-N-1}{k-j-1} \sum_{p=0}^{\infty} \frac{j^p}{p!} \frac{(-N)_p}{(-N-k+j+1)_p} \\ &= \binom{-N-1}{k-1}\mu^N + \mu^N \sum_{j=1}^{k-1} \binom{k}{j} \binom{-N-1}{k-j-1} M(-N, -N-k+j+1, j) \end{aligned}$$

Finally we compute G_k^N for $N < 0$ and $k > -N$. By the definition of S_k^N above, we can write

$$\begin{aligned}
G_k^N = &\sum_{j=1}^{k+N-1} \sum_{p=N+k-j}^{\infty} \sum_{\mathbf{x}:x_1+\ldots+x_j=p} \frac{\mu^p}{\prod_{t=1}^{j} x_t!} \frac{1}{\mu^{p-N}} \binom{k}{j} \binom{p-N-1}{k-j-1} \\
&+ \sum_{j=k+N}^{K-1} \sum_{p=0}^{\infty} \sum_{\mathbf{x}:x_1+\ldots+x_j=p} \frac{\mu^p}{\prod_{t=1}^{j} x_t!} \frac{1}{\mu^{p-N}} \binom{k}{j} \binom{p-N-1}{k-j-1} \\
&= \mu^N \sum_{j=1}^{k+N-1} \binom{k}{j} j^{N+k-j} \; \mathbf{M}(k-j, N+k-j+1, j) \\
&\quad + \mu^N \sum_{j=k+N}^{k-1} \binom{k}{j} \binom{-N-1}{k-j-1} M(-N, -N-k+j+1, j)
\end{aligned}$$

Proof of Lemma 4

We proceed by computing the total throughput T_K of the servers which can be rewritten as:

$$\begin{aligned}
X_K = \frac{\mu}{G_K} \Bigg(&\sum_{j=2}^{K-1} \sum_{n=K-j}^{\infty} \sum_{m=1}^{n+1} \frac{1}{m!} \cdot \sum_{\mathbf{x}:x_1+\ldots+x_{j-1}=n-m+1} \prod_{t=1}^{j-1} \frac{1}{x_t!} j \binom{K}{j} \binom{n-1}{K-j-1} \\
&+ \sum_{n=K-1}^{\infty} \frac{1}{(n+1)!} K \binom{n-1}{K-2} + K \\
&+ \sum_{j=1}^{K-1} \sum_{n=K-j}^{\infty} \sum_{\mathbf{x}:x_1+\ldots+x_j=n} \prod_{t=1}^{j} \frac{1}{x_t!} (K-j) \binom{K}{j} \binom{n-1}{K-j-1} \Bigg),
\end{aligned}$$

where $(x_1, \ldots, x_j)$ is a possible tuple of the non-negative components of state $\mathbf{n}$, such that $\sum_{i=1}^{j} x_i = n$, with $2 \leq j \leq K-1$. If x_j has value $m-1$ then the remaining $x_1, \ldots, x_{j-1}$ sum to $n-m+1$. There are j possible ways of inserting x_j into the sequence $x_1, \ldots, x_{j-1}$ to form a j-tuple of non-negative components and $\binom{k}{j}$ possible ways of assigning $(x_1, \ldots, x_j)$ to the values of a state $\mathbf{n} \in \mathcal{S}$. The remaining $(K-j)$ components are negative. The cases $j = 1$ and $j = K$ are treated separately. By applying the multinomial theorem we obtain:

$$\begin{aligned}
X_K = \frac{\mu}{G_K} \Bigg(&\sum_{j=2}^{K-1} \sum_{n=K-j}^{\infty} \sum_{m=1}^{n+1} \frac{1}{m!} \frac{(j-1)^{n-m+1}}{(n-m+1)!} \\
&\cdot j \binom{K}{j} \binom{n-1}{K-j-1} + \sum_{n=K-1}^{\infty} \frac{1}{(n+1)!} K \binom{n-1}{K-2} + K \\
&+ \sum_{j=1}^{K-1} \sum_{n=K-j}^{\infty} \frac{(j)^n}{n!} (K-j) \binom{K}{j} \binom{n-1}{K-j-1} \Bigg).
\end{aligned}$$

and by the binomial formula:

$$\sum_{m=1}^{n+1} \binom{n+1}{m} (j-1)^{n-m+1} = j^{n+1} - (j-1)^{n+1}.$$

Hence, the throughput can be written as:

$$\begin{aligned} X_K = \frac{\mu}{G_K} \Bigg(& \sum_{j=2}^{K-1} \sum_{n=K-j}^{\infty} \frac{j}{(n+1)!} \binom{K}{j} \binom{n-1}{K-j-1} \\ & \cdot (j^{n+1} - (j-1)^{n+1}) + \sum_{n=K-1}^{\infty} \frac{1}{(n+1)!} K \binom{n-1}{K-2} + K \\ & + \sum_{j=1}^{K-1} \sum_{n=K-j}^{\infty} \frac{(j)^n}{n!} (K-j) \binom{K}{j} \binom{n-1}{K-j-1} \Bigg). \end{aligned}$$

The rest of the proof is purely algebraic. Indeed, from the above expression we can derive:

$$\begin{aligned} X_K = \frac{\mu}{G_K} \Bigg(& \sum_{j=2}^{K-1} \binom{K}{j} j^{K-j+2} \sum_{w=0}^{\infty} \frac{j^w \ (K-j)_w}{\Gamma(w+K-j+2) \ w!} \\ & - \sum_{j=2}^{K-1} \binom{K}{j} j(j-1)^{K-j+1} \sum_{w=0}^{\infty} \frac{(j-1)^w \ (K-j)_w}{\Gamma(w+K-j+2) \ w!} + \\ & + K \sum_{w=0}^{\infty} \frac{(K-1)_w}{\Gamma(w+K+1) \ w!} + K \\ & + \sum_{j=1}^{K-1} \binom{K}{j} (K-j) j^{K-j} \sum_{w=0}^{\infty} \frac{j^w \ (K-j)_w}{\Gamma(w+K-j+1) \ w!} \Bigg). \end{aligned}$$

and by Equation (7), we have:

$$\begin{aligned} X_K = \frac{\mu}{G_K} \Bigg(& \sum_{j=2}^{K-1} \binom{K}{j} j^{K-j+2} \ \mathbf{M}(K-j, K-j+2, j) \\ & - \sum_{j=2}^{K-1} \binom{K}{j} j(j-1)^{K-j+1} \ \mathbf{M}(K-j, K-j+2, j-1) \\ & + K \ \mathbf{M}(K-1, K+1, 1) + K \\ & + \sum_{j=1}^{K-1} \binom{K}{j} (K-j) j^{K-j} \ \mathbf{M}(K-j, K-j+1, j) \Bigg) \end{aligned}$$

Now, observe that since $\mathbf{n}(t)$ is ergodic and the join operation is instantaneous we have that the expected join-queue length must be finite. Since by symmetry

the throughput of each server must be the same, i.e., X_K/K and there is not an infinite accumulation of customers in the join-queue, the throughput X is X_K/K. □