Steady-state approximations of dynamic speed-scaling in data centers

B.J. Prabhu*† A.E. Tugui ‡ I.M. Verloop§¶
* CNRS, LAAS, 7 avenue du colonel Roche, F-31400 Toulouse, France
† Université de Toulouse, LAAS, F-31400 Toulouse, France
‡ Military Technical Academy, Bucarest, Romania
§ CNRS-IRIT, Toulouse, France
¶ Université de Toulouse, INP-ENSEEIHT, France

***Abstract*—Dynamic speed-scaling, which varies the server speed with the number of tasks, has been proposed to balance energy and delay costs in data-centers. The direct numerical computation of these costs under the optimal speed-scaling policy does not give much insight into their behavior. By making an analogy with the Erlang-C model, we propose approximations for the mean energy consumed per task and the mean delay in systems with dynamic speed-scaling. These approximations are related to those by Halfin and Whitt for the Erlang-C system. The applicability of these approximations is illustrated with the help of comparison with the exact optimal routing policy in data-centers.**

***Index Terms*—energy efficiency, data-centers, Erlang-C, Gaussian approximation**

I. Introduction

Energy consumption of computer and communication system has been increasing rapidly in the last few years. A non-negligible proportion of this consumption can be attributed to data-centers which are large clusters of processors that store and process information. The energy consumption of a data center can be as high as that of a small city requiring many megawatts of power. In 2010, the energy consumed by data centers worldwide was estimated to be between one and one and a half percent of the worldwide electricity-based energy consumption [1]. As energy costs increase, data-center service providers are seeking to reduce energy consumption without overly compromising on the Quality-of-Service.

A possible way to reduce energy consumption is to vary the speed of the processors in the data-center. It has been observed that the power consumed by a server operating at a speed s is polynomial in s, that is

$$P \propto s^{\alpha}.$$

where $\alpha = 3$ for CMOS processors [2]. Thus, a lower operating speed means a lower energy consumption. Unfortunately, this reduction in energy cost has a direct adverse impact on system performance thereby implying a trade-off between the amount of saved energy and the loss in system performance.

This trade-off has been investigated in both the static-setting in which there is a finite number of tasks (*e.g.* [2], [3], and references therein) and in the stochastic setting in which there are an infinite number of tasks that arrive according to a stochastic process (see, *e.g.*, [4], [5] and references therein).

If the objective function is a linear combination of the mean energy per task and mean sojourn time per task, then it has been shown that the policy

$$s \overset{n\to\infty}{\longrightarrow} cn^{1/\alpha},$$

where n is the number of tasks in the system, is asymptotically optimal in the stochastic setting with PS scheduling policy [4], and 2-competitive in the static setting with SRPT scheduling policy [3]. Such a policy in which the speed of the server is varied as a function of the number of tasks is called *dynamic speed-scaling*.

Instead of the exact optimal policy, which requires numerical computation, one could employ the policy:

$$s = cn^{1/\alpha}, \tag{1}$$

which is ready-to-use and is optimal when there are a large number of tasks in the system. It has the added advantage of a closed-form expression for the steady-state distribution of the number of tasks when the system can be modelled as an $M/G/1/PS$ queue. We shall refer to this policy as *dynamic speed-scaling*.

Once the server-speed policy has been determined, the following problems arise: *(i)* determine the performance (for example, mean sojourn time of tasks, mean energy consumed per task) for a given load; and *(ii)* determine the optimal routing probabilities in a data-center with heterogeneous servers.

Consider an $M/M/1$ queue in which tasks arrive at rate λ. Tasks volumes are assumed to be exponential with mean 1 unit. The server has a maximum speed of S unit of work per unit time, that is the server can execute at most S tasks per unit time on an average.

Let $\pi_\alpha(n)$ be the steady-state number of tasks in the system with dynamic speed-scaling. From standard Markov chain analysis, it follows that

$$\pi_\alpha(n) = \pi_\alpha(0)\frac{\lambda^n}{\prod_{i=1}^{n}\min(i^{1/\alpha}, S)}, \tag{2}$$

where $\pi_\alpha(0) = \left[\sum_{n=0}^{\infty}\frac{\lambda^n}{\prod_{i=1}^{n}\min(i^{1/\bullet}, S)}\right]^{-1}$.

It has been assumed that the constant in (1) is equal to 1. This assumption corresponds to a change in the units of λ and S. The steady-state distribution of the queue with

parameters (λ, S, c) is the same as that of the queue with parameters $(\lambda/c, S/c, 1)$. So, there is no loss in generality in this assumption.

Let $\bar{N}_\alpha(\lambda, S)$ (resp. $\bar{P}_\alpha(\lambda, S)$) denote the mean number of tasks (resp. mean power consumed) in the system. Then,

$$\bar{N}_\alpha(\lambda, S) = \sum_{n=0}^{\infty} n\pi(n), \tag{3}$$

$$\bar{P}_\alpha(\lambda, S) = \sum_{n=0}^{\infty} \min(n, S^\alpha)\pi(n). \tag{4}$$

The mean sojourn time per task and the mean energy consumed per task can now be computed using Little's law.

At this point, we stop to make the following observation: for $\alpha = 1$ and $S \in \mathbb{Z}_+$, the system under consideration is just the $M/M/S$ queueing system introduced by Erlang. While there are analytical expression for the steady-state distribution and mean number of tasks, the need for approximations even for this case was expressed by Erlang himself. Since then, several approximations have been proposed for the computation of the mean number of tasks and the probability of a task having to wait (see, *e.g.* [6], [7], [8]).

In this paper, we shall give approximations for the mean number of tasks and the mean power consumed for a server with dynamic speed-scaling. The approximations are related to those in Halfin and Whitt [7]. Although, the approximations proposed by Janssen *et al.* [8] are sharper than those in [7], they rely on the possibility of computing the blocking probability using the incomplete gamma function. For $\alpha \neq 1$, to the best of the authors' knowledge, there is no such representation. Therefore, we fall back on the usual saddle point and Striling's approximations used in Halfin and Whitt.

The applications of these approximations are two-fold : *(i)* compute the desired performance measures for a single server queue and compare them with static-speed policies; and *(i)* compute the optimal Markovian routing policies for a farm of servers which are dynamically speed-scaled.

The rest of this paper is organized as follows. In Section II, approximations for the mean number of tasks, variance of the number of tasks and the probability of the system being empty are derived for a server with infinite maximum speed. In Section III, we compute the approximations for a server with finite maximum speed. These approximations are similar to the ones computed in [7] for $\alpha = 1$. In Section IV, first we numerically compare these approximations with the corresponding exact values. Then, we provide a comparison between the energy and delay costs obtained by computing the optimal routing policy using the approximations and the exact formula.

II. The infinite maximum server speed case

For $\alpha = 1$, the system is analogous to the $M/M/\infty$ which has the Poisson distribution for the steady-state number of tasks with mean number of tasks being λ. For other values of α, no similar closed-form expression for these quantities are known (at least to the authors). In such cases, one resorts to approximations for different sets of the parameter values. One could restrict the analysis for particular values of the arrival rate, for example small values of λ (or, light-traffic approximations) or large values of λ (or, heavy traffic approximations). In the rest of the note, we shall be interested in the latter approximation.

The power consumed when the maximum speed is infinite is the same as the mean number of tasks in the system. Therefore, we shall concentrate on approximations for the latter. The main result of this section is:

Theorem 2.1: For $S = \infty$, as $\lambda \to \infty$,

$$\pi_\alpha(0) \to (2\pi)^{-\left(\frac{1}{2}-\frac{1}{2\alpha}\right)} \alpha^{-\frac{1}{2}} \lambda^{-(\alpha-1)/2} \exp\left(-\frac{\lambda^\alpha}{\alpha}\right), \tag{5}$$

and

$$\bar{N}_\alpha(\lambda, \infty) - \lambda^\alpha \to \frac{\alpha - 1}{2}. \tag{6}$$

Moreover, the variance of the number of tasks is given by

$$\mathrm{Var}(N_\alpha(\lambda, \infty)) = \alpha\lambda^\alpha. \tag{7}$$

In order to prove Theorem 2.1, it will be easier to work with

$$f_\alpha(\lambda, c) \equiv \sum_{n=0}^{\lambda^\alpha + c(\alpha\lambda^\alpha)^{1/2}} \frac{\lambda^n}{(n!)^{1/\alpha}}. \tag{8}$$

Note that, $\pi_\alpha(0) = f_\alpha(\lambda, \infty)^{-1}$, and

$$\bar{N}_\alpha(\lambda, \infty) = \lambda f_\alpha(\lambda, \infty)^{-1} \frac{df_\alpha(\lambda, \infty)}{d\lambda}, \tag{9}$$

$$\mathrm{Var}(N_\alpha(\lambda, \infty)) = \lambda^2 f_\alpha(\lambda, \infty)^{-1} \frac{d^2 f_\alpha(\lambda, \infty)}{d\lambda^2} - \bar{N}_\alpha(\lambda, \infty)^2 + \bar{N}_\alpha(\lambda, \infty). \tag{10}$$

so that the desired result can be deduced by approximating $f_\alpha(\lambda, \infty)$, which is given in the following result.

Lemma 2.1:

$$f_\alpha(\lambda, c) \overset{\lambda\to\infty}{\longrightarrow} (2\pi)^{\frac{1}{2}-\frac{1}{2\alpha}} \alpha^{\frac{1}{2}} \lambda^{(\alpha-1)/2} \exp\left(\frac{\lambda^\alpha}{\alpha}\right) \Phi(c), \tag{11}$$

where Φ is the cumulative distribution function of the standard Normal distribution.

The proof of the above lemma is omitted due to lack of space. It relies on the classical method of supposing that a few terms around the maximum are the main contributors towards the sum. The peripheral terms can then be neglected, and the restricted sum is approximated by an integral to arrive at the desired approximation.

We can make the following observations related to Theorem 2.1.

1) As $\lambda \to \infty$, the mean number of tasks grows as λ^α. While this is the right order of growth in the heavy-traffic regime, it is unfortunately not the case in the light-traffic regime. To see this, approximate $\pi_\alpha(0)$ by its first two terms in the sum, that is

$$\pi_\alpha(0) = \left(1 + \lambda + \frac{\lambda^2}{2^{1/\alpha}}\right)^{-1} \approx 1 - \left(\lambda + \lambda^2\left(\frac{1}{2} + \frac{1}{2^{1/\alpha}}\right)\right).$$

In the expression for $\bar{N}_\alpha(\lambda,\infty)$, we retain the terms of the order of λ^2 and neglect the other terms to obtain

$$\bar{N}_\alpha(\lambda,\infty) = \pi_\alpha(0)\left(\lambda + \tfrac{2}{2^{1/\alpha}}\lambda^2\right) \approx \lambda + \lambda^2(2^{1-1/\alpha}-1).$$

From which we can deduce that $\lim_{\lambda\to 0}\frac{\bar{N}_\alpha(\lambda,\infty)}{\lambda} = 1$. For $\alpha > 1$, since $\lambda^\alpha = o(\lambda)$ as $\lambda \to 0$, the relative error[1] between the approximation and the exact value will tend to 1.

Note that for $\alpha = 1$, the approximation is in fact the exact value for all values of λ. This highlights one of specificities of $\alpha \neq 1$. The approximations obtained would not be well-suited for small values of λ.

2) The term $\frac{\alpha-1}{2}$, though superfluous when λ is large, comes into play for moderate values of λ. For this reason, we choose to keep this term rather than approximating with just λ^α. In Section IV, examples will be given to illustrate the usefulness of this term.
The upside of keeping this term is that the approximation is better for a larger range of λ. On the downside, for smaller values of λ the approximation becomes negative.
3) For $\lambda < 1$, $\bar{N}_\infty(\lambda,\infty) = \frac{\lambda}{1-\lambda}$. The proposed approximation does not arrive at this value for $\alpha = \infty$. It is either 0 when $\lambda < 1$ (and the $(\alpha-1)/2$ is not considered), 1 if $\lambda = 1$, or it is ∞ when $\lambda > 1$.

III. The finite maximum server speed case

First, we shall compute an approximation for the mean number of tasks given in (3). Following Halfin and Whitt [7], divide the sum (3) into two parts:

$$\bar{N}_\alpha(\lambda,S) = \pi_\alpha(0)\sum_{n=0}^{S^\alpha} n\frac{\lambda^n}{(n!)^{1/\alpha}} + \pi_\alpha(0)\sum_{n=S^\alpha+1}^{\infty} n\frac{\lambda^n}{(S^\alpha!)^{1/\alpha}S^{n-S^\alpha}}.$$

While for them the first term in the sum has a rather simple form due to α being 1, in our case no such simplification results. After some algebra, we arrive at

$$\bar{N}_\alpha(\lambda,S) = (1-\gamma)\frac{\sum_{n=0}^{S^\alpha} n\frac{\lambda^n}{(n!)^{1/\alpha}}}{\sum_{n=0}^{S^\alpha}\frac{\lambda^n}{(n!)^{1/\alpha}}} + \gamma\left(\frac{\rho}{1-\rho} + S^\alpha + 1\right),$$

where

$$\gamma = \textstyle\sum_{n=S^\alpha+1}^{\infty}\pi_\alpha(n) = \pi_\alpha(0)\frac{\lambda^{S^\alpha}}{(S^\alpha!)^{1/\alpha}}\frac{\rho}{1-\rho}, \qquad (12)$$

and $\rho = \lambda/S$.

As a first approximation, the first term on the right-hand side can be substituted by $(1-\gamma)\bar{N}_\alpha(\lambda,\infty)$. An approximation for $\bar{N}_\alpha(\lambda,\infty)$ was already derived in (6). So, let us turn our attention to γ.

Let $c = \frac{S^\alpha-\lambda^\alpha}{(\alpha\lambda^\alpha)^{1/2}}$. Assume that as $\lambda\to\infty$, c tends to a strictly positive finite value.

Lemma 3.1:

$$\gamma \overset{\lambda\to\infty}{\longrightarrow} \left(1 + \sqrt{2\pi}c\exp\left(\frac{c^2}{2}\right)\Phi(c)\right)^{-1} \qquad (13)$$

[1] We define the relative error between approximation (a) and the exact value (e) as $(e-a)/e$.

Combining the results in Theorem 2.1 and Lemma 2.1, the following approximations for the mean number of tasks and the mean power consumed are obtained.

Proposition 3.1: Let $c = (S^\alpha - \lambda^\alpha)/(\alpha\lambda^\alpha)^{1/2}$ be strictly positive and finite. As $\lambda\to\infty$,

$$\bar{N}_\alpha(\lambda,S) \approx (1-\gamma)\left(\lambda^\alpha + \tfrac{\alpha-1}{2}\right) + \gamma\left(\tfrac{\lambda}{S-\lambda} + S^\alpha + 1\right), \qquad (14)$$

$$\bar{P}_\alpha(\lambda,S) \approx (1-\gamma)\left(\lambda^\alpha + \tfrac{\alpha-1}{2}\right) + \gamma S^\alpha, \qquad (15)$$

with γ as defined in (13).

For $\alpha = 1$, Halfin and Whitt proposed the following approximation :

$$\bar{N}_1(\lambda,S) \approx \lambda + \gamma\left(\tfrac{\lambda}{S-\lambda}\right), \qquad (16)$$

where γ and c are as defined previously with $\alpha = 1$. The approximation in (14) is not quite the same as (16). There is a difference of $O(\lambda^{\alpha/2})$, which means that asymptotically they are the same but for smaller values of λ they could be different. The approximation (16) can be interpreted as the sum of two terms, the first of which is the mean number of the infinite server case. The second term corresponds to the additional number of tasks due to the server operating at its maximum speed (i.e., the $M/M/1$ case) weighted by the probability that the server is operating at its maximum speed. If we naively follow this logic, then we arrive at the approximation

Proposition 3.2:

$$\bar{N}_\alpha(\lambda,S) \approx \left(\lambda^\alpha + \tfrac{\alpha-1}{2}\right) + \gamma\left(\tfrac{\lambda}{S-\lambda}\right), \qquad (17)$$

$$\bar{P}_\alpha(\lambda,S) \approx (1-\gamma)\left(\lambda^\alpha + \tfrac{\alpha-1}{2}\right) + \gamma S^\alpha, \qquad (18)$$

with γ as defined in (13).

IV. Numerical results

A. Single Processor case

In Table I, we compare the exact mean number of tasks (*cf.* (3)) and the approximation (*cf.* Theorem 2.1) for the infinite maximum speed case. For $\lambda > 1$, the approximation is reasonably good whereas for $\lambda < 1$ the approximation has a much higher error. For fixed λ, this error increases as α increases. This behaviour was already predicted in Section II,

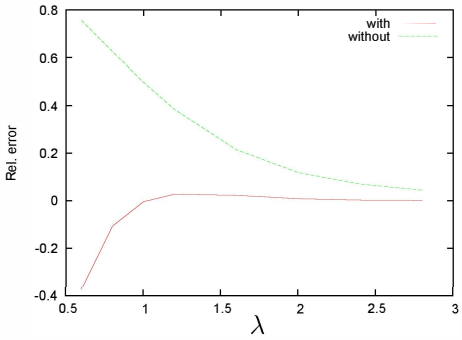
TABLE I: Exact and approximate mean number of tasks for $S = \infty$.

λ	$\alpha = 2$			$\alpha = 3$		
	Exact	Approx.	Rel. error.	Exact	Approx.	Rel. error.
0.6	0.772	0.860	-0.112	0.885	1.216	-0.373
0.8	1.122	1.140	-0.016	1.366	1.512	-0.106
1.0	1.526	1.500	0.017	1.991	2.000	-0.004
1.2	1.992	1.940	0.026	2.805	2.728	0.027
1.6	3.125	3.060	0.020	5.214	5.096	0.022
2.0	4.554	4.500	0.011	9.075	9.000	0.008
2.4	6.298	6.260	0.006	14.860	14.824	0.002
2.8	8.365	8.340	0.003	22.970	22.952	0.0008

where it was noted that for $\alpha = \infty$, the approximation predicts $-\infty$, whereas the exact value is $\lambda/(1-\lambda)$.

Before moving on to the finite maximum speed case, in Figure 1 we illustrate the usefulness of the term $(\alpha-1)/2$ in (6). It can be seen that the approximation with $(\alpha-1)/2$ is better over a much larger range of λ.

Fig. 1: Relative error in the approximation with and without $(\alpha-1)/2$ for $\alpha = 3$.



For the finite maximum speed case, we compare the two approximations (14) and (17) with the exact values. In Table II, the exact values and the relative error for the two approximations are given for $S = 3$. They are already reasonably good even for $\rho = 0.2$. It is observed though that the approximation (17) is better for values of ρ close to 1. We also mention that

TABLE II: Comparison of approximations for $S = 3$.

ρ	$\alpha = 2$			$\alpha = 3$		
	Exact (3)	Rel. err. (14)	Rel. err. (17)	Exact	Rel. err. (14)	Rel. err. (17)
0.20	1.288	0.1128	-0.1128	1.475	-3.73e-01	-3.73e-01
0.40	1.661	0.0268	0.0268	2.337	2.74e-02	2.74e-02
0.60	2.148	0.0056	0.0276	3.851	1.45e-02	1.45e-02
0.80	3.104	0.0915	0.0299	6.264	-1.32e-02	7.70e-03
0.90	4.794	0.0737	0.0192	8.560	-6.64e-02	1.04e-02
0.95	8.130	0.0345	0.0102	12.11	-5.30e-02	6.49e-03
0.99	34.79	0.0041	0.0020	38.89	-5.31e-03	8.67e-04

the approximations improve as the value of S increases, and that the approximations for the mean power consumed have a similar relative error.

B. Optimal routing for data-centers

Consider a set of J servers with S_j being the maximum speed of server j. Tasks arrive according to a Poisson process of rate Λ and are routed probabilistically to a server as soon as they arrive. The objective is to determine the routing probabilities so as to minimize the function

$$\sum_{j=1}^{S} \bar{N}_\alpha(\lambda_j, S_j) + \beta \bar{P}_\alpha(\lambda_j, S_j)$$

with the constraint that $\sum_j \lambda_j = \Lambda$.

Intuitively, at low loads the maximum speed of the servers will be attained with very low probability. The servers will appear to be homogeneous with respect to their maximum speeds. Therefore, it will be optimal to route equal amounts of traffic to the servers.

As the system load will increase, the performance of a server will be determined by its maximum speed (the $M/M/1$ term in (14)) and the optimal policy will start to resemble the water-filling policy for $M/G/1/PS$ server farms.

This intuition is confirmed in Table III, in which we present the optimal policies computed by optimising over the exact objective function and over the two approximations. In the last column, the value of the exact objective function obtained with these policies is also given.

TABLE III: Comparison of policies and objective function. $J = 5$, $[S_1, S_2, S_3, S_4, S_5] = [2, 3, 4, 5, 6]$, $\alpha = 3$.

ρ		λ_1^*	λ_2^*	λ_3^*	λ_4^*	λ_5^*	Value
0.05	Exact (3)	0.2	0.2	0.2	0.2	0.2	2.258
	App. (14)	0.2	0.2	0.2	0.2	0.2	2.258
	App. (17)	0.2	0.2	0.2	0.2	0.2	2.258
0.5	Exact (3)	1.489	1.627	1.627	1.627	1.627	6.57
	App. (14)	1.393	1.651	1.651	1.651	1.651	6.591
	App. (17)	1.439	1.64	1.64	1.64	1.64	6.576
0.95	Exact (3)	1.877	2.836	3.777	4.636	4.872	35.117
	App. (14)	1.883	2.847	3.773	4.529	4.966	35.19
	App. (17)	1.881	2.842	3.777	4.573	4.925	35.144

Even though at low traffic, the approximations have a large relative error, the optimal policy obtained by optimising over the approximations has the same form as that obtained from the exact functions. This is due to the fact that in both the approximations and the exact function, at low arrival rates, the servers appear homogeneous, and thus they have the same optimal policy. This rather fortunate coincidence implies that near-optimal routing can be expected for a larger range of traffic-intensities than the validity of the approximations themselves.

V. Acknowledgment

This work was partially supported by the ANR SOP project ANR-11-INFR-001.

References

[1] J. G. Koomey, *Growth in Data Center Electricity used 2005 to 2010*. New York, NY, USA: The New York Times, August 1, 2010. [Online]. Available: JGKoomey@stanford.edu, http://www.koomey.com

[2] N. Bansal, T. Kimbrel, and K. Pruhs, "Speed scaling to manage energy and temperature," *J. ACM*, vol. 54, no. 1, pp. 3:1–3:39, Mar. 2007.

[3] L. L. H. Andrew, A. Wierman, and A.Tang, "Optimal speed scaling under arbitrary power functions," *SIGMETRICS Perform. Eval. Rev.*, vol. 37, no. 2, pp. 39–41, oct 2009.

[4] A. Wierman, L. L. H. Andrew, and A. Tang, "Power-aware speed scaling in processor sharing systems," in *Proc. IEEE INFOCOM*, Rio de Janeiro, Brazil, 20-25 Apr 2009, pp. 2007–2015.

[5] L. L. H. Andrew, M. Lin, and A. Wierman, "Optimality, fairness and robustness in speed scaling designs," in *Proc. ACM SIGMETRICS*, New York, NY, 14-18 Jun 2010, pp. 37–48.

[6] B. Halachmi and W. R. Franta, "A diffusion approximation to the multi-server queue," *Management Science*, vol. 24, no. 5, pp. pp. 522–529, 1978.

[7] S. Halfin and W. Whitt, "Heavy-traffic limits for queues with many exponential servers," *Operations Research*, vol. 29, no. 3, pp. pp. 567–588, 1981.

[8] A. J. E. M. Janssen, J. S. H. v. Leeuwaarden, and B. Zwart, "Refining square-root safety staffing by expanding erlang c," *Operations Research*, vol. 59, no. 6, pp. pp. 1512–1522, 2011.