Optimizing the Number of Robots for Web Search Engines

Jérôme Talim – Zhen Liu – Philippe Nain
INRIA, B.P. 93, 06902, Sophia Antipolis Cedex, France
{jtalim, liu, nain}@sophia.inria.fr

E. G. Coffman, Jr.
Bell Labs, Lucent Technologies
Murray Hill, NJ 07974, USA
egc@bell-labs.com

May 29, 1998

Abstract

Robots are deployed by a Web search engine for collecting information from different Web servers in order to maintain the currency of its data base of Web pages. In this paper, we investigate the number of robots to be used by a search engine so as to maximize the currency of the data base without putting an unnecessary load on the network. We use a queueing model to represent the system. The arrivals to the queueing system are Web pages brought by the robots; service corresponds to the indexing of these pages. The objective is to find the number of robots, and thus the arrival rate of the queueing system, such that the indexing queue is neither starved nor saturated. For this, we consider a finite-buffer queueing system and define the cost function to be minimized as a weighted sum of the loss probability and the starvation probability. Under the assumption that arrivals form a Poisson process, and that service times are independent and identically distributed random variables with an exponential distribution, we obtain explicit solutions for the optimal number of robots to deploy.

Keywords: Web search engines, Web robots, Queueing system

1 Introduction

The World Wide Web has become a major information publishing and retrieving mechanism on the Internet. The amount of information as well as the number of Web servers has been growing exponentially fast in recent years. In order to help users find useful information on the Web, search engines such as Alta Vista, HotBot, Yahoo, Infoseek, Magellan, Excite and Lycos, etc. are available. These systems consist of four main components: a database that contains web pages (full text or summary), a user interface that deals with queries, an indexing engine that updates the database, and robots that traverse the Web servers and bring Web pages to the indexing engine. Thus, the quality of a search engine depends on many factors, e.g., query response time, completeness, indexing speed, currency, and efficient robot scheduling.

Our interest here focuses on the function served by robots: establishing currency by bringing new pages to be indexed and bringing changed/updated pages for re-indexing. We investigate the problem of choosing the number of robots to meet the conflicting demands of low network traffic and an up-to-date data base. The specific model, illustrated in Figure 1, centers on the indexing engine, which is represented by a finite, single-server queue/buffer, and multiple robots acting as sources of arriving pages. The times between successive page accesses are independent and identically distributed for each robot; the robots themselves are identical and function independently. The indexing (service) times are independent, identically distributed, and independent of the arrival processes.

When a robot arriving with a page for the indexing buffer finds the buffer full, the page being delivered is lost, at least temporarily. In this situation, a potential update or new page has been lost, and network congestion has been created to no benefit. On the other hand, if the buffer is ever empty, and hence the indexing engine is idle, data base updating is at a standstill waiting for the robots to bring more pages. To reduce the probability of the first of these two events, we want to keep the number of robots suitably small, but to reduce the probability of the second, we want to keep the number of robots suitably large. To make the objective concrete, we will formulate a cost function as a weighted sum of the probabilities of an empty buffer and a full buffer. We will then study the problem of finding the number of robots that minimizes the cost function.

There is a large literature on search engines and their components. The search engines themselves may well be their own best source of references; we recommend this entree to the research on any aspect of the subject. In particular, much can be found on the design and control (including distributed control) of robots. However, we have found very little on the modeling and analysis of robot scheduling and the indexing queue. The work in [2] is the only such effort we know about. In [2], the authors propose a natural model of Web-page obsolescence, and study the problem of scheduling a single search engine robot so as to minimize the extent to which the search engine's data base is out-of-date.

Section 2 introduces the probability model, sets notation, and formalizes the optimization problem. Section 3 solves the optimization problem and presents an explicit computation of the optimal number of robots. The sensitivity of the results to model parameters is also studied in some detail. Section 4 concludes with a brief discussion of further, more general results not included here and of interesting questions that remain open.

2 Preliminaries and general notation

The size of the finite buffer is K; we assume that $K \geq 2$. The number of robots is denoted by N. Each is a Poisson source with rate λ; the sources are independent of each other. Then for N robots, the total arrival rate is $N\lambda$. As noted earlier, we assume that the arrival and service processes are independent.

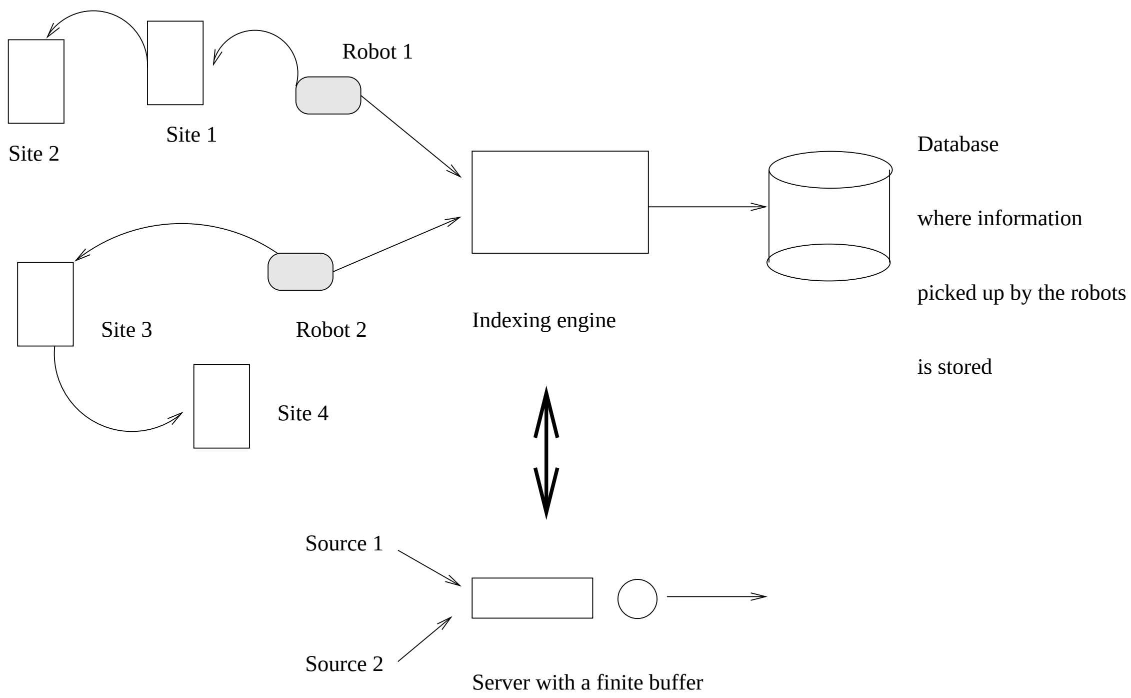


Figure 1: Model of search engine

Based on these definitions, we define the cost function as follows. As mentioned earlier, it is the weighted sum of two terms :

- The probability of an empty buffer $\mathbf{P}\{X = 0\}$, where X is a random variable with the stationary queue-length distribution.
- The probability of losing an arriving page. This is the probability that the queue length seen by an arrival is K, which we denote by $\mathbf{P}^*\{X = K\}$

If we define $\rho := N\lambda/\mu$, then the cost function can be written as:

$$C(\rho, \gamma, K) := \gamma \mathbf{P}\{X = 0\} + \mathbf{P}^*\{X = K\}$$

where γ is a positive constant that is useful when we want to stress one term or the other.

We use the results of queueing theory to compute $C(\rho, \gamma, K)$ (see e.g., [3]). In the calculation of the value of ρ that achieves $\min_\rho C(\rho, \gamma, K)$, we first consider ρ as a continuous variable. And then, depending on the behavior of the cost function, we will derive the optimal (discrete) number of robots.

3 The M/M/1/K search engine model

By the assumptions of Section 2, our search engine model is the well-known M/M/1/K queue. We first address the problem of finding the optimal number of sources in this model.

3.1 Optimizing the number of robots

The following proposition gives the well known stationary queue-length probabilities at arbitrary epochs for this queue [3]:

Proposition 1 *For any $\rho > 0$,*

$$\mathbf{P}\,(X = i) = \begin{cases} \dfrac{1-\rho}{1-\rho^{K+1}}\rho^i & \textit{for } i = 0, 1, \ldots, K \\ 0 & \textit{for } i > K. \end{cases}$$

When $\rho = 1$ the non-zero stationary queue-length probabilities at arbitrary epochs are all equal and given by $Prob\,(X = i) = 1/(K+1)$ for $i = 0, 1, \ldots, K$.

The expression for the cost function $C(\rho, \gamma, K)$ now flows from Proposition 1 and the PASTA property that ensures that in the M/M/1/K queue the stationary queue-length probabilities at arbitrary epochs and the stationary queue-length probabilities at arrival epochs are equal (i.e., $\mathbf{P}^*\,(X = k) = \mathbf{P}\,(X = k)$). We find that

$$C(\rho, \gamma, K) = \begin{cases} \dfrac{(1-\rho)(\gamma + \rho^K)}{1-\rho^{K+1}} & \text{for } \rho \neq 1 \\ \dfrac{\gamma + 1}{1 + K} & \text{for } \rho = 1. \end{cases} \tag{1}$$

Note that for any $K \geq 2$, $\gamma > 0$, the mapping $\rho \to C(\rho, K, \gamma)$ is continuous and differentiable at each point in $(0, \infty)$, including the point $\rho = 1$.

Lemma 1 *For any $\gamma > 0$, $K \geq 2$, the mapping $\rho \to C(\rho, \gamma, K)$ has a unique minimum in $[0, \infty)$, to be denoted $\rho(\gamma, K)$. Furthermore, $0 < \rho(\gamma, K) < 1$ if $\gamma < 1$, $\rho(1, K) = 1$ and $\rho(\gamma, K) > 1$ if $\gamma > 1$.*

Proof. Fix $K \geq 2$. We have

$$\frac{\partial C(\rho, \gamma, K)}{\partial \rho} = \frac{R(\rho, \gamma, K)}{(1-\rho^{K+1})^2} \tag{2}$$

with

$$R(\rho, \gamma, K) = \rho^{2K} - K\gamma\rho^{K+1} + (\gamma - 1)(K+1)\rho^K + K\rho^{K-1} - \gamma. \tag{3}$$

Tedius but elementary algebra shows that the polynomial $R(\rho, \gamma, K)$ in the variable ρ

(i) has a zero of multiplicity two (respectively, three) at point $\rho = 1$ when $\gamma \neq 1$ (respectively, $\gamma = 1$);

(ii) has a zero of multiplicity one in $[0, 1)$ and no zero in $(1, \infty)$ when $\gamma < 1$;

(iii) has a zero of multiplicity one in $(1, \infty)$ and no zero in $[0, 1)$ when $\gamma > 1$;

(iv) has no zero other than 1 when $\gamma = 1$.

We deduce from the above that

$$\frac{\partial C(\rho,\gamma,K)}{\partial \rho} = \frac{(1-\rho)^2}{(1-\rho^{K+1})^2}\, Q(\rho,\gamma,K)$$

where $Q(\rho,\gamma,K)$ is a polynomial in the variable ρ with a single zero $\rho(\gamma,K)$ in $[0,\infty)$ with $\rho(\gamma,K) < 1$ if $\gamma < 1$, $\rho(1,K) = 1$ and $\rho(\gamma,K) > 1$ if $\gamma > 1$. Furthermore, the inequality $Q(0,\gamma,K) = -\gamma < 0$ implies that, for any $\gamma > 0$, $Q(\rho,\gamma,K) < 0$ for $0 \le \rho < \rho(\gamma,K)$ and $Q(\rho,\gamma,K) > 0$ for $\rho > \rho(\gamma,K)$, which proves the lemma.

♡

It is worth observing that the optimum $\rho(\gamma,K)$ does not depend on the buffer size K when $\gamma = 1$. This means that if the same weight is given to the probability of starvation and to the loss probability, then the optimal arrival rate is equal to the service rate, independent of the buffer size.

We now return to the original problem, namely the computation of the number N of robots that minimizes the cost function $C(\rho,\gamma,K)$ with $\rho = \lambda N/\mu$. The answer is found in the next result which is a direct corollary of Lemma 1.

Proposition 2 *For any $\gamma > 0$, $K \ge 2$, let $N(\gamma,K)$ be the optimal number of robots to use.*

Then,

$$N(\gamma,K) = \arg \min_{n\in\{\lfloor \rho(\gamma,K)\mu/\lambda\rfloor, \lceil \rho(\gamma,K)\mu/\lambda\rceil\}} C(n\lambda/\mu,\gamma,K) \tag{4}$$

where for any real number x, $\lfloor x \rfloor$ (respectively $\lceil x \rceil$) denotes the largest (respectively smallest) integer less (respectively more) than or equal to x.

In the next section, we investigate the impact of the parameter γ on the optimal number of robots.

3.2 Impact of γ on the optimal number of robots

Recall that the parameter γ is a positive constant that allows us to stress either the loss probability or the probability of starvation. Part of the impact of γ on $\rho(\gamma,K)$, and therefore on $N(\gamma,K)$, the optimal number of robots, is captured in the following result.

Proposition 3 *For any $\gamma > 0$, $K \ge 2$, the mapping $\rho \to \rho(\gamma,K)$ is nondecreasing in $[0,\infty)$, with $\lim_{\gamma\to\infty} \rho(\gamma,K) = \infty$.*

Proof. Pick two constants $0 < \gamma_1 < \gamma_2$ and define

$$\begin{aligned}\Delta(\rho,\gamma_1,\gamma_2,K) &:= C(\rho,\gamma_2,K) - C(\rho,\gamma_1,K)\\ &= \frac{1-\rho}{1-\rho^{K+1}}\,(\gamma_2 - \gamma_1.)\end{aligned}$$

Let us assume that $\rho(\gamma_2,K) < \rho(\gamma_1,K)$ and show that this leads to a contradiction.

The mapping $\rho \to \Delta(\rho,\gamma_1,\gamma_2,K)$ is stricly decreasing in $[0,\infty)$. Therefore,

$$\begin{aligned} 0 &< \Delta(\rho(\gamma_2,K),\gamma_1,\gamma_2,K) - \Delta(\rho(\gamma_1,K),\gamma_1,\gamma_2,K)\\ &= [C(\rho(\gamma_2,K),\gamma_2,K) - C(\rho(\gamma_1,K),\gamma_2,K)] + [C(\rho(\gamma_1,K),\gamma_1,K) - C(\rho(\gamma_2,K),\gamma_1,K)] \le 0 \end{aligned} \tag{5}$$

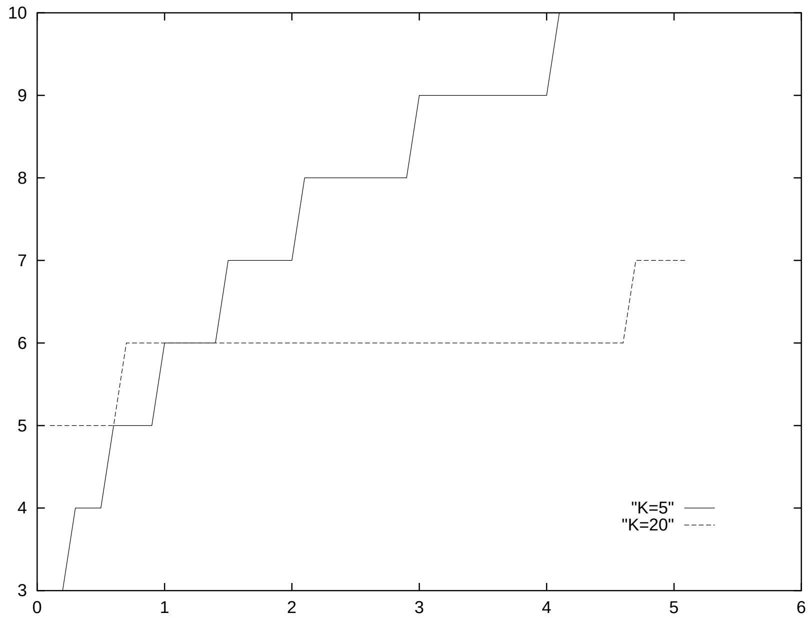


Figure 2: Optimal number of robots as a function of γ, with $\mu/\lambda = 5.7$

where the last inequality follows from the definition of $\rho(\gamma, K)$. This ends up as a contradiction, which proves the first part of the proposition.

On the other hand, it is easily checked that $\partial C(\rho, \gamma, K)/\partial \rho < 0$ for $\rho = (\gamma/K)^{1/(K-1)}$ when $\gamma > 0$. Hence, by Lemma 1, we deduce that, necessarily,

$$\rho_0(\gamma, K) := \left(\frac{\gamma}{K}\right)^{1/(K-1)} < \rho(\gamma, K), \quad \forall \gamma > 0. \tag{6}$$

Letting γ tend to infinity on both sides of (6) yields the second result of the proposition.

♡

Proposition 3 has a simple physical interpretation. As the parameter γ increases the probability of starvation becomes the main quantity to minimize. The minimization is done by increasing the arrival rate or, equivalently, by increasing the number of robots. Figure 2 provides two numerical examples, illustrating the monotonicity of the optimal number of robots. One should note that the rate of increase of this function seems to decrease when the buffer size increases.

The next section focuses on the impact of the buffer size K on the optimal number of robots.

3.3 Impact of K on the optimal number of robots

In this section, we examine the behavior of $\rho(\gamma, K)$ as a function of K. The first result establishes an upper bound on $\rho(\gamma, K)$ that complements the lower bound given in (6).

Lemma 2 *For any $\gamma > 0$, $K \geq 2$,*

$$\rho(\gamma, K) < ((K+1)\gamma)^{1/(K-1)} := \rho_1(\gamma, K). \tag{7}$$

Proof. Thanks to Lemma 1, it is enough to show that $\partial C(\rho,\gamma,K)/\partial\rho > 0$ at the point $\rho = \rho_1(\gamma,K)$ or, equivalently from (2), that $R(\rho_1(\gamma,K),\gamma,K) > 0$.

By writing $R(\rho,\gamma,K)$ in the form

$$R(\rho,\gamma,K) = \rho^{K+1}\left(\rho^{K-1} - K\gamma\right) + (\gamma-1)(K+1)\rho^K + K\rho^{K-1} - \gamma,$$

we find that

$$R(\rho_1(\gamma,K),\gamma,K) = ((K+1)\gamma))^{\frac{K+1}{K-1}} + (\gamma-1)(K+1)\left((K+1)\gamma\right)^{\frac{K}{K-1}} + \gamma(K(K+1)-1)$$

which is strictly positive, in particular for $K \geq 2$ and $\gamma > 0$.

♡

The lower and upper bounds (6) and (7) combine to yield

$$\lim_{K\to\infty} \rho(\gamma,K) = 1 \tag{8}$$

for any $\gamma > 0$. In other words, the optimal arrival rate converges to the service capacity when the buffer size increases to infinity.

In terms of the optimal number of robots to be used when $K \to \infty$, we see from (8) and (4) that

$$\lim_{K\to\infty} N(\gamma,K) = \lim_{K\to\infty} \arg \min_{\{n\in\lfloor\mu/\lambda\rfloor,\lceil\mu/\lambda\rceil\}} C(\lambda\, n/\mu,\gamma,K). \tag{9}$$

Hence, for K large enough, (9) suggests the approximation

$$N(\gamma,K) \sim \begin{cases} \lceil\mu/\lambda\rceil & \text{if } C(\rho_+,\gamma,\infty) \leq C(\rho_-,\gamma,\infty) \\ \lfloor\mu/\lambda\rfloor & \text{if } C(\rho_+,\gamma,\infty) \geq C(\rho_-,\gamma,\infty) \end{cases}$$

or, equivalently from (1),

$$N(\gamma,K) \sim \begin{cases} \lceil\mu/\lambda\rceil & \text{if } (\rho_+ - 1)/\rho_+ \leq \gamma\,(1-\rho_-) \\ \lfloor\mu/\lambda\rfloor & \text{if } (\rho_+ - 1)/\rho_+ \leq \gamma\,(1-\rho_-) \end{cases} \tag{10}$$

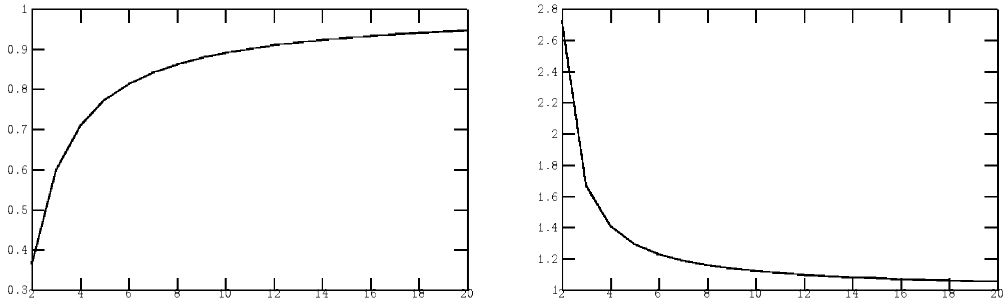
with $\rho_+ := (\lambda/\mu)\lceil\mu/\lambda\rceil$ and $\rho_- := (\lambda/\mu)\lfloor\mu/\lambda\rfloor$.

The limiting result (8) may seem counterintuitive at first. Indeed, one may be tempted to argue that the component $P(X=K)$ in the cost function $C(\rho,\gamma,K)$ converges to 0 as the buffer size increases to infinity and to conclude that $C(\rho,\gamma,K)$ is minimized when $P(X=0)$ converges to 0, which occurs when the arrival rate converges to infinity.

This intepretation is not correct as $\lim_{K\to\infty} P(X=K) = (\rho-1)/\rho > 0$ when $\rho > 1$ (see Proposition 1).

It is easily checked from Proposition 1 that $\lim_{\rho\to 1}\lim_{K\to\infty} C(\rho,\gamma,K) = \lim_{K\to\infty}\lim_{\rho\to 1} C(\rho,\gamma,K) = 0$, which agrees with (8).

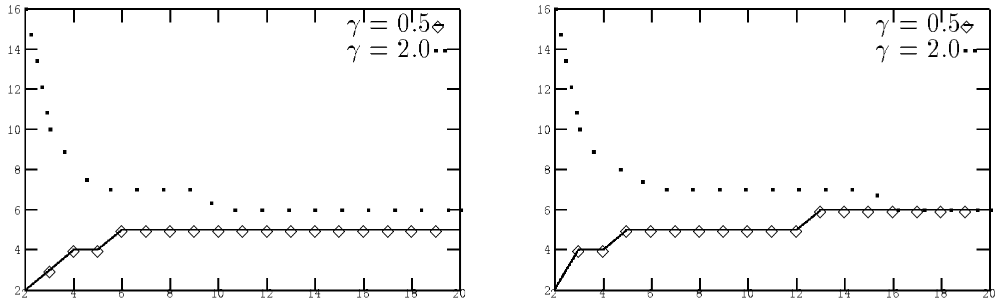
It is not an easy task to study the behavior of $\rho(\gamma,K)$ as a function of K. We suspect the mapping $K \to \rho(\gamma,K)$ to be increasing when $0 < \gamma < 1$ and decreasing when $\gamma > 1$, but we have not been able to prove it. The conjectured behavior of the mapping $K \to \rho(\gamma,K)$ (and of $K \to N(\gamma,K)$) is illustrated in Figures 3 and 4. Figure 5 illustrates the behavior of the optimal number of robots as a function of the ratio μ/λ, when the buffer size is quite large. In both curves, the parameter γ is fixed (equals 0.5 and 2.0).



(a) $\rho(\gamma, K)$ for $\gamma = 0.5$

(b) $\rho(\gamma, K)$ for $\gamma = 2$

Figure 3: The mapping $K \mapsto \rho(\gamma, K)$



(a) $N(\gamma, K)$ for $\frac{\mu}{\lambda} = 5.7$

(b) $N(\gamma, K)$ for $\frac{\mu}{\lambda} = 6.0$

Figure 4: The mapping $K \mapsto N(\gamma, K)$

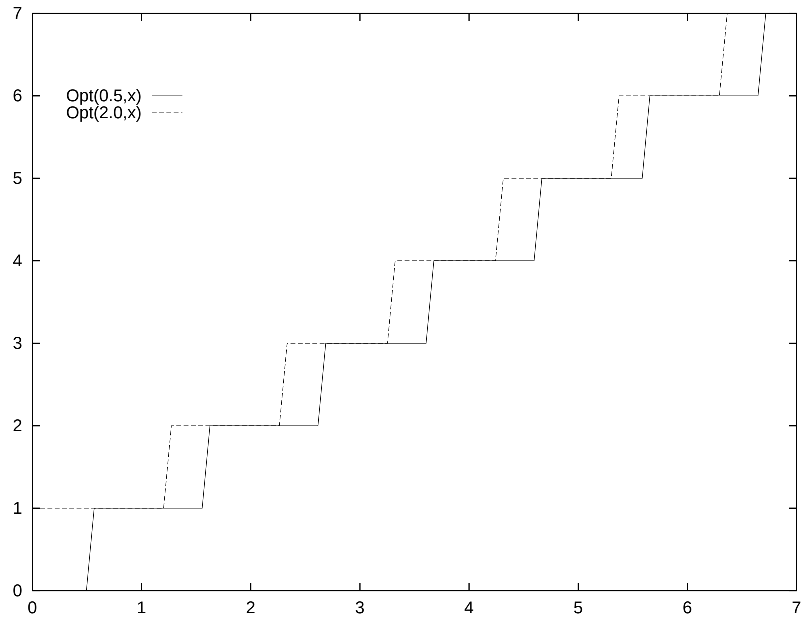


Figure 5: Optimal number of robots as a function of μ/λ, for "large" buffer

4 Final Remarks

Useful generalizations are obtained by replacing exponential distributions by general ones. We have done this in part by solving our optimization problem for a broad class of indexing-time distributions. This analysis will appear in an expanded version of the paper.

The reader may have already noticed that a search engine with a variable, dynamically changing number of active robots/sources could lead to better solutions in general. For example, one can contemplate robot-control rules that deactivate robots as the buffer approaches the full state, and activate more robots as the buffer approaches the empty state. The authors have a companion paper in progress on this extension [4].

Finally, a realistic model may require that robots not all be considered identical. They may operate in different geographical neighborhoods, for example, in which case our problem could become part of a larger problem in which the optimal location of robots is also included.

References

[1] J. W. Cohen. *The Single Server Queue.* North-Holland Publishing Company, 1982.

[2] E. G. Coffman, Jr., Zhen Liu, and Richard R. Weber. Optimal robot scheduling for web search engines. *J. Scheduling*, (in press).

[3] L. Kleinrock, *Queueing Systems, Vol. I.* Wiley & Sons, New York, 1975.

[4] J. Talim, Z. Liu, P. Nain, and E. G. Coffman, Jr. Optimal number of robots for search engines: dynamic case. *preprint*, October 1997.

[5] E. B. Saff and A. D. Snider. *Complex Analysis for Mathematics, Science, and Engineering.* Prentice Hall, Englewood Cliffs, 1982.