Optimizing the number of robots for web search engine

Govind Kothari

Department of Computer Science and Engineering, IIT Guwahati

Roll no. 04010114, email: govind@iitg.ernet.in

Abstract

With the rise in the amount of web data, it is not possible to search for needed information manually. Therefore search engines are used. Robots are part of web search engine which automatically crawl the web. These crawled (downloaded) pages are fed to the indexer which indexes them for later use of sending results when the user sends a query. A queuing model is used to represent such system. Arrivals to the queuing model are web pages send crawled by the robots, service corresponds o indexing these pages. This report investigates the optimum number of robots that should be used by search engine, and hence the arrival rate of the queuing system, so as to maintain the freshness of its database without increasing the network load. If the number of robots increases, it may so happen the indexer gets overloaded and crawled pages are dropped, if it becomes less the indexer may get starved. A finite-buffer queuing model is used. A cost function is defined, which is the weighted sum of the loss probability and starvation probability. Our aim is to minimize this cost function. Under the assumption that the arrivals form a Poisson process and service times are independent and identically distributed, we try to find out numerical solution to optimum number of robot to deploy.

1. Introduction

Web search engine consist of three parts:

1) Robot (crawler, spider): for downloading web pages
2) Indexer: parses the web page and stores information about that page(key words, URL etc)
3) Web server: to serve user queries

Robots act as a feeder to the indexer. We can model this as a queuing system, where indexer is the server which serves the arriving web pages sent by robot. Now in our problem we have a single server (i.e. one indexer) and multiple robots. Server has a finite buffer, which can be thought as a finite queue length. The times between successive page accesses by each robot are independent and identically distributed. The robots themselves are identical and function independently. The indexing times are independent and identically distributed and independent of the arrival process.

Now if the robot arrives with a page and finds the buffer full, the page being delivered is dropped or lost. This results in unnecessary congestion of network bandwidth. But at the same time if we use less robots to avoid the above situation, it may so happen that the indexer runs out of pages and remains idle. This leads to under utilization of resources. Thus we require number of robots such that there is balance between the two situations. Thus we define a cost function which is the weighted sum of the two situations and try to minimize this cost function.

2. The Working Model

The search engine is modeled as a single server finite capacity queue. The system capacity is $K \geq 2$. There are $N \geq 1$ robots each bringing pages according to Poisson process with rate $\lambda > 0$.These Poisson process are mutually independent and independent of service time. Thus the net arrival rate is λN. It is obtained by taking N random variable with each with arrival rate λ and taking the minimum of that. An incoming page finding a full queue is lost.

Let us denote $F(x) = \mathbf{P}\{\sigma \leq x\}$ the probability distribution function of the service times and $\sigma' > 0$ as mean service time. We define $\mu = 1/\sigma'$.

Thus we define cost function as the weighted sum of two terms:

1) The probability that buffer is empty $\mathbf{P}\{X= 0\}$ where X is the random variable representing the stationary queue length in M/G/1/K system with arrival rate λN and service time distribution F.
2) The probability of loosing the arriving page i.e. when the queue is full which we denote as $\mathbf{P}^*\{X=K\}$

With $\rho= N\lambda/\mu$, the cost function is

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

where γ is a positive constant denoting weight.

Thus our aim is to find the N such that $C(\rho,\gamma,K)$ is minimum.

In the paper both M/M/1/K and M/G/1/K model is given. I will resort to only the first one in this report.

3. The M/M/1/K Model

3.1. Optimizing the number of robots

We know that in M/M/1/K system, the stationary queue length probability at arbitrary epochs is given by

$P\{X=i\} = \frac{1-\rho}{1-\rho k+1}\rho^{i}$ *for i=0,1…,K*

0 *for i>K*

When $\rho = 1$ the non zero queue length at arbitrary epoch are all equal and given by

$P\{X=i\} = 1/(K+1)$ *for i=0,1,…,K*

The cost function is given by

$C(\rho,\gamma,K) = \frac{(1-\rho)(\gamma+\rho k)}{1-\rho k+1}$ *for* $\rho\neq 1$

$\frac{\gamma+1}{1+k}$ *for* $\rho=1$

It should be noted that for any $K \geq 2$, $\gamma>0$, the mapping $\rho \rightarrow C(\rho,\gamma,K)$ is continuous and differentiable in $(0,\infty)$,including the point $\rho=1$.

Lemma 1: *For any γ>0 , K≥2, the mapping ρ→ C(ρ,γ,K) has a unique minimum in [0,∞), to be denoted as ρ(γ,K).Furthermore, 0<ρ(γ,K)<1 if γ<1, ρ(1,K)=1 and ρ(γ,K)>1 if γ>1.*

This can be proved by differentiating the cost function and looking various cases. Observation of its proof is that the optimum ρ(γ,K) does not depend on K when γ=1. The straight inference from this is that if the equal weight is given to loss and starvation probability than optimal arrival rate is equal to service rate (ρ=1), independent of buffer size (K).

Now we come back to finding=g optimum N.

Proposition1: *For any γ>0, K≥2, let N(γ,K)be the optimal number of robots to use. Then*

$$N(\gamma,K)= \mathrm{argmin}_{n} C(n\lambda/\mu,\gamma,K)$$

with $n \in \{\lfloor \rho(\gamma,K)\mu/\lambda \rfloor, \lceil \rho(\gamma,K)\mu/\lambda \rceil\}$

3.2. Effect of γ on number of robots (N)

Proposition2: *For any $K \geq 2$, the mapping $\gamma \rightarrow \rho(\gamma,K)$ is non decreasing in $[0,\infty)$ with $\lim_{\gamma\rightarrow\infty} \rho(\gamma,K)=\infty$.*

From lemma1 and proposition2 we get

$\rho_0(\gamma,K):= (\frac{\gamma}{K})^{1/(K-1)} \leq \rho(\gamma,K) \quad , \forall\gamma > 0$ **(1)**

This proposition has simple physical interpretation. If increase γ, the starvation probability is main component to be minimized and this requires increase in the number of robots.

3.3. Effect of K on number of robots (N)

Lemma: *For any $\gamma \geq 1$, $K \geq 2$*

$\rho(\gamma,K)< ((K+1)\gamma)^{1/(k-1)} := \rho_1(\gamma,K)$ **(2)**

Using lemma1 and lower and upper bounds on *ρ(γ,K)*,as in equation (1) and (2), we have

$\rho_0(\gamma,K) \leq \rho(\gamma,K) <1 ,$ *for* $0<\gamma<1$ **(3)**

and

$\rho_1(\gamma,K) \geq \rho(\gamma,K)$ *for* $\gamma>1$ **(4)**

By combining (3) and (4) with the limits we get

$\lim_{K\rightarrow\infty} \rho(\gamma,K) = 1$ *for* $\gamma>0$.

Thus the optimal arrival rate converges to service capacity when buffer size increases to infinity.

From various approximations it has been observed that N(γ,∞) is 6 when γ lies between {0.5,2).

4. Conclusion

From simple queuing model (M/M/1/K) of search engines we were able to get the optimum number of robots to be used. From the idea of this model we can also work for the case where the number of robots are dynamic i.e. they can increase or decrease with requirement.

There are various open issues brought to light like where robots are non homogeneous, existing at different locations of network. Like one may look for optimum number of robots to be allocated to particular locality.