Journal of Universal Computer Science, vol. 12, no. 9 (2006), 1139-1153
submitted: 31/12/05, accepted: 12/5/06, appeared: 28/9/06 © *J.UCS*

Performance Modeling of Proxy Cache Servers

[1] **Tamás Bérczes, János Sztrik**
(Department of Informatics Systems and Networks, University of Debrecen
P.O. Box 12, H-4010, Debrecen, Hungary
{tberczes, jsztrik}@inf.unideb.hu)

Abstract: The primary aim of the present paper is to modify the performance model of Bose and Cheng [1] to a more realistic case when external visits are also allowed to the remote Web servers and the Web servers have limited buffer. We analyze how many parameters affect the performance of a Proxy Cache Server. Numerical results are obtained for the overall response time with and without a PCS. These results show that the benefit of a PCS depends on various factors. Several numerical examples illustrate the effect of visit rates, visit rates for the external users , and the cache hit rate probability on the mean response times.

Key Words: Queueing Network, Proxy Cache Server, Performance Models
Category: H.1.1, H.3.3

1 Introduction

The World Wide Web (WWW) can give a quick and easy access to a large number of web servers where users can find all kind of information, documents and multimedia files. ¿From the user's point of view it does not matter whether the requested files are on the firm's computer or on the other side of the world. The usage of the web has been growing very fast. The number of internet users increased from 474 million in 2001 to 590 million in 2002, and the forecast for 2006 is 948 million users. According to the facts, that in 1996 the number of users was only 627000, the growth is rapid and we can justify and exponential grows in traffic, too. The users want to get a high quality service and modest response time. The answer from the remote web server to the client often takes a long time. One of the problems is that the same copy of the file can be claimed by other users at the same time. Because of this situation, identical copies of many files pass through the same network links, resulting in an increased response time. A natural solution to avoid this situation is to store this information. In general caching can be implemented at browser software; the originating Web sites; and the boundary between the local area network and the Internet. Browser cache are inefficient since they cache for only one user. The caching at the Web sites can improve performance, although the requested files are still subject to delivery through the Internet. It has been suggested that the greatest improvement in response time for corporations will come from installing a proxy cache server (PCS) at the boundary between the local area network and the Internet. Requested documents can be delivered directly from the web server or through

[1] Research is partially supported by the Hungarian National Foundation for Scientifc Research under grant OTKA-K60698.

a proxy cache server. A PCS has the same functionality as a web server when looked at from the client and the same functionality as a client when looked at from a web server. The primary function of a proxy cache server is to store documents close to the users to avoid retrieving the same document several times over the same connection.

In this paper a modification of the performance model of Bose and Cheng [1] is given to deal with a more realistic case when external visits are also allowed to the remote Web servers and the Web servers have a limited buffer. For the easier understanding of the basic model and comparisons we follow the structure of the cited work. In Section 2 we construct a queuing network model to study the dynamics of installing a PCS. Overall response-time formulas are developed for both the case with and without a PCS. In Section 3 numerical experiments are conducted to examine the response time behavior of the PCS with respect to various parameters of the model. Concluding remarks can be found in Section 4.

2 An analytical model of Proxy Cache Server traffic

In this section we briefly describe the mathematical model with the suggested modifications. Using proxy cache server, if any information or file is requested to be downloaded, first it is checked whether the document exists on the proxy cache server. (We denote the probability of this existence by p). If the document can be found on the PCS then its copy is immediately transfered to the user. In the opposite case the request will be sent to the remote Web server. After the requested document arrived to the PCS then the copy of it is delivered to the user.

The advantage of a PCS depends on several factors: The probability of the "cache hit rate" of the PCS, the speed of the PCS, the bandwidth of the firm's and the remote network and the speed of the remote web server [1].
Fig. 1 illustrates the path of a request in the modified model starting from the user and finishing with the return of the answer to the user. The notations used in this model are collected in Table 1.
We assume that the requests of the PCS users arrive according to a Poisson process with rate λ, and the external visits at the remote web server form a Poisson process with rate Λ.

Let F be the average of the requested file size. We define λ_1, λ_2, λ_3 and λ_5 such that:

$$\lambda_1 = p * \lambda \text{ and } \lambda_2 = (1-p) * \lambda \tag{1}$$

$$\lambda_3 = \lambda_2 + \Lambda \text{ ,} \lambda_5 = (1 - P_b) * \lambda_2 \tag{2}$$

The solid line in Fig 1. (λ_1) represents the traffic when the requested file is available on the PCS and can be delivered directly to the user. The λ_2 traffic depicted by dotted

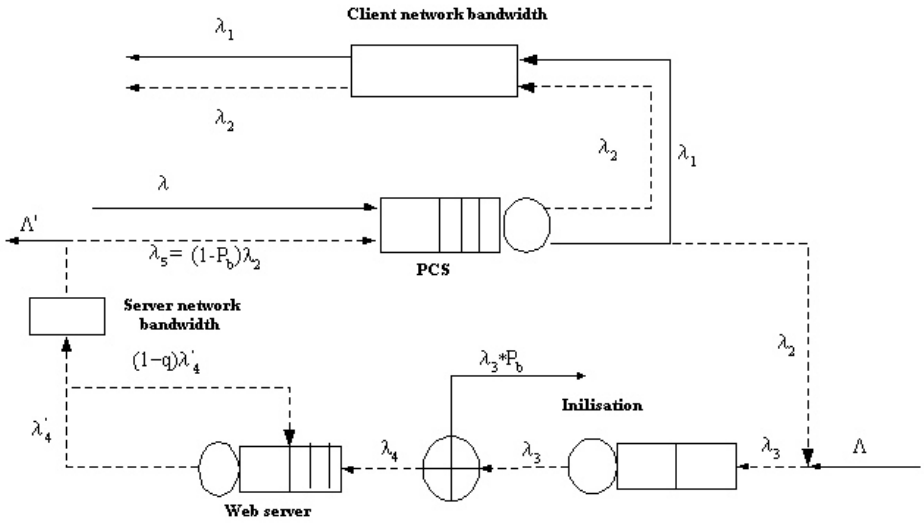

Figure 1: Network model

line, represents those requests which could not be served by the PCS, therefore these requests must be delivered from the remote web server. Naturally the web server serves not only the requests of the studied PCS but it also serves requests of other external users. Let λ_3 denote the intensity of the overall requests arriving to the remote Web server. The λ_3 traffic undergoes the process of initial handshaking to establish a one-time TCP connection [7], [1]. We denote by I_s this initial setup.

According to [1], "The remote Web server performance is characterized by the capacity of its output buffer B_s, the static server time Y_s, and the dynamic server rate R_s." In our model we assume that the Web server has a buffer of capacity K. Let P_b be the probability that a request will be denied by the Web server. As it is well-known from basic queueing theory the blocking probability P_b for the $M/M/1/K$ queueing system:

$$P_b = P(N = K) = \frac{(1-\rho) * \rho^K}{1-\rho^{K+1}} \tag{3}$$

where

$$\mu = \frac{R_s B_s}{F(Y_s R_s + B_s)} \tag{4}$$

Now we get

$$\rho = \frac{\lambda_3 F(Y_s R_s + B_s)}{R_s B_s} \tag{5}$$

Now we can see that the requests arrive to the buffer of the Web server according to a Poisson process with rate

$$\lambda_4 = (1 - P_b) * \lambda_3 \tag{6}$$

The performance of the firm's PCS is characterized by the parameters B_{xc}, Y_{xc} and R_{xc}.
If the size of the requested file is greater then the Web server's output buffer it will start a looping process until the delivery of all requested file's is completed. Let

$$q = \min\left(1, \frac{B_s}{F}\right) \tag{7}$$

be the probability that the desired file can be delivered at the first attempt. Let $\lambda_4^{'}$ be the rate of the requests arriving at the Web service considering the looping process. According to the conditions of equilibrium and the flow balance theory of queueing networks

$$\lambda_4 = q * \lambda_4^{'} \tag{8}$$

Then, we get the overall response time:

$$\begin{aligned} T_{xc} = {} & \frac{1}{\frac{1}{I_{xc}} - (\lambda)} + p * \left\{ \frac{1}{\frac{B_{xc}}{F*\left(Y_{xc}+\frac{B_{xc}}{R_{xc}}\right)} - \lambda_1} + \frac{F}{N_c} \right\} \\ & + (1-p) * \left\{ \frac{1}{\frac{1}{I_s} - \lambda_3} + \frac{1}{\frac{B_s}{F*\left(Y_s+\frac{B_s}{R_s}\right)} - \frac{\lambda_4}{q}} \right. \\ & \left. + \frac{F}{N_s} + \frac{1}{\frac{B_{xc}}{F*\left(Y_{xc}+\frac{B_{xc}}{R_{xc}}\right)} - \lambda_5} + \frac{F}{N_c} \right\}, \end{aligned} \tag{9}$$

The response time T_{xc} consists of tree terms.
The first term is the time to check whether the requested file is on the PCS or not. This is derived form the waiting time in an $M/M/1$ queueing system where the visits form a Poisson process with rate λ and the service rate is $\frac{1}{I_{xc}}$.
The second term is the response time in the case if the requested document exists on the PCS, the probability of which is p. The first item in this term is the waiting time on the PCS, where the numerator $\frac{B_{xc}}{F*\left(Y_{xc}+\frac{B_{xc}}{R_{xc}}\right)}$ is the "service demand". The second item in the second term correspond to the required time for content to travel through the client network bandwidth.
The third term is the response time when the requested file does not exist on the PCS. The probability of that event is $(1-p)$. This term consists of tree terms too. The first item is the expected one-time initialization time of the TCP connection between the PCS and the remote web server. The second item is the waiting time of the queueing system on the remote Web server, where $\lambda_4/q = \lambda_4^{'}$ and F/N_s is the expected time of transferring the requested documents on the network of the bandwidth. The third term

is the waiting time of the PCS when the copy of the requested document is transfered to the user.
When there is no PCS, the overall response time T, is given by the same arguments:

$$T = \frac{1}{\frac{1}{I_s} - (\lambda + \Lambda)} + \frac{1}{\frac{B_s}{F*\left(Y_s + \frac{B_s}{R_s}\right)} - \frac{(1-P_b)*(\lambda+\Lambda)}{q}} + \frac{F}{N_s} + \frac{F}{N_c} \tag{10}$$

3 Numerical results

For the numerical explorations the corresponding parameters of Cheng and Bose [1] are used. The value of the other parameters for numerical calculationd are: $I_s = I_{xc} = 0.004$ seconds, $B_s = B_{xc} = 2000$ bytes, $Y_s = Y_{xc} = 0.000016$ seconds, $R_s = R_{xc} = 1250$ Mbyte/s, $N_s = 1544$ Kbit/s, and $N_c = 128$ Kbit/s.
In Figures 2.- 10. the dotted line plot the case with a PCS and the normal line depicts the case without a PCS.

3.1 Effect of visit rate

In Fig 2. the response time is depicted as a function of the visit rate. In this Figure the visit rates for the external users is 100 requests/s and the cache hit rate is 0.1. We see that in this case the response time will be greater when we install a PCS. In Fig 3. we use the same parameters, but the cache hit rate is 0.25. In this case the response time is the same with and without a PCS. In Fig. 4 we use a higher visit rate for the external users ($\Lambda = 150$) with a smaller cache hit rate ($p = 0.1$). When λ is smaller then 70 requests/s the response time is larger with a PCS than without a PCS. When we use a higher cache hit rate with a higher visit rate for the external users (Fig 5, $p = 0.25$, $\Lambda = 150$) the efficiency of PCS is clear. In this case the response time with a PCS will be smaller than the response time without a PCS for any value of the visit rate. So, we can see that the performance of a PCS depends on a high scale of the firms behaviour, but when the intensity of the requests from the firm is greater than 70, and the visit rate for the external users is 150 requests/s than it is enough a small cache hit rate to access a smaller response time.

3.2 Effect of visit rates for the external users

Now we investigate the effect of the visit rate for the external users . In Fig. 6 the visit rate from the PCS is 20 requests/s, the requested file size is 5000 byte and the cache hit rate is 0.1. We can see that with these parameters installing a PCS we get a higher response time. In Fig. 7 we modified only the cache hit rate probability to 0.25. In

this situation when we have more than 140 external requests/s then the response time with PCS is smaller than without a PCS. When the cache hit rate probability is smaller ($p = 0.1$) and $\lambda = 70$ requests/s (Fig. 8) then the response time with a PCS is smaller than without, when we use a greater visit rate for the external users ($\Lambda > 150$). When we use a higher cache hit rate probability (Fig. 9 $p = 0.4$) then the response time with a PCS is smaller, independently of the external visits. Observing Fig. 6 - 9, we can find that in general the response time with and without a PCS increases when the visit rate for the external users increases. When the visit rate of the studied firm is modest (20 requests/s) then the benefit of the PCS is visible when the visit rates for the external users are bigger or when the cache hit rate probability is higher.

3.3 Comparison of the two model

When the visit rate for the external users is zero ($\Lambda = 0$) and the the buffer size (K) for the remote Web server is unlimited we get the equation used by Bose and Cheng. Fig 10. ($p = 0.1$, $F = 5000$ bytes, $\Lambda = 0$) depicts the overall response time as a function of the arrival rate given by the equation used by Bose and Cheng and Fig 4. depicts the response time given by our model with the same parameters using visit rate for external users ($\Lambda = 150$) and buffer size ($K = 100$). Investigating Fig 4. we see that the response time with PCS will be smaller than without a PCS only in the case when we use a higher visit rate ($\lambda > 85$). In Fig 4. the advantage of the PCS will be visible with lower visit rates ($\lambda > 65$). The numerical examples show us that using our modifications we got a more realistic queueing network model. Using our modified model when the external visits are allowed, the PCS was beneficial with a lower traffic.

3.4 Effect of the cache hit rate probability

Fig. 11 ($\lambda = 20$, $\Lambda = 100$, $F = 5000$ bytes, K=100) depicts the overall response time as a function of the cache hit rate probability. The overall response time without a cache server is independent of p, as observed from (10). Hence we use $T \approx 0.321$s for all values of p. The overall response time with a PCS decreases as p increases as seen in Fig. 11. When we use a cache hit rate probability ≈ 0.2 the efficiency of a PCS is visible.

3.5 Effect of the buffer size

Let T_{xc}^{∞} denote the response time, when we use unlimited buffer for the Web server. In this case the blocking probability P_b is 0, so from (6) we get that $\lambda_4 = \lambda_3$. We can easily derive the overall response time using equation (9):

$$T_{xc}^{\infty} = \frac{1}{\frac{1}{T_{xc}} - (\lambda)} + p * \left\{ \frac{1}{\frac{B_{xc}}{F*\left(Y_{xc}+\frac{B_{xc}}{R_{xc}}\right)} - \lambda_1} + \frac{F}{N_c} \right\}$$
$$+ (1-p) * \left\{ \frac{1}{\frac{1}{I_s} - \lambda_3} + \frac{1}{\frac{B_s}{F*\left(Y_s+\frac{B_s}{R_s}\right)} - \frac{\lambda_3}{q}} \right. \tag{11}$$
$$\left. + \frac{F}{N_s} + \frac{1}{\frac{B_{xc}}{F*\left(Y_{xc}+\frac{B_{xc}}{R_{xc}}\right)} - \lambda_2} + \frac{F}{N_c} \right\},$$

In Fig 12., 13. we depict the response time in function of the buffer size. In both Figures the dotted line plot the response time when we use Web server with limited buffer and the normal line illustrates the case when we use unlimited buffer. Of course, using unlimited buffer we get constant value for the response time. Accordingly, we would like to get information, regarding the effect of the buffer size on the response time.

When the visit rate for the external users is zero ($\Lambda = 0$) then the response time given by the equation equals to the response time given by the equation that was used by Bose and Cheng. In Fig 12. ($p = 0.1, \lambda = 70, \Lambda = 0, F = 5000$ bytes) we find that the buffer size of the Web server effects the response time only if it is very small. When the buffer size (K) is more than 5 , the response time is independent of the buffer size. In Fig. 13 we analyze the situation when external visits are allowed ($p = 0.1, \lambda = 70, \Lambda = 100$, $F = 5000$ bytes). We can observe that allowing external visits the effect of the buffer size vanishes at a bigger value. In our case this happens when K is greater than 8.

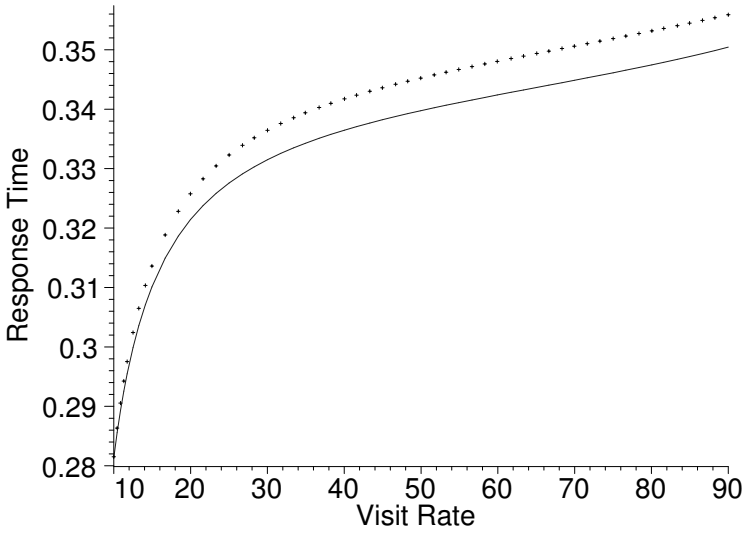

Figure 2: $p = 0.1$, $F = 5000$ bytes,$\Lambda = 100$, K=100

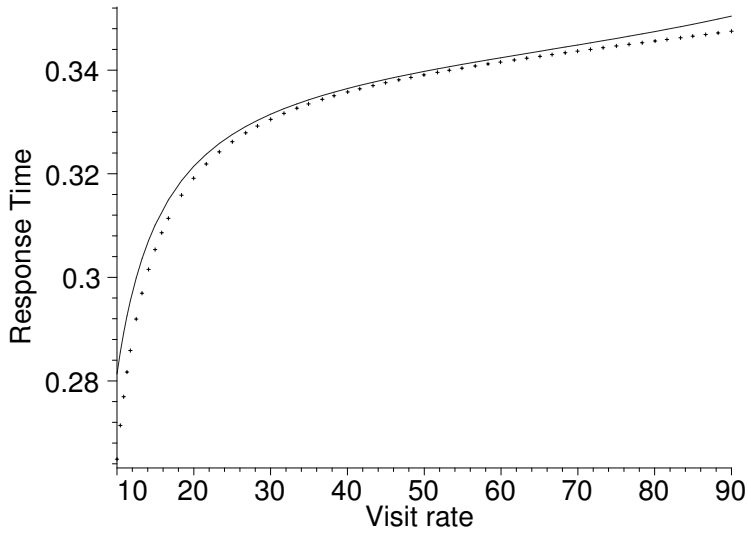

Figure 3: $p = 0.25$, $F = 5000$ bytes,$\Lambda = 100$, K=100

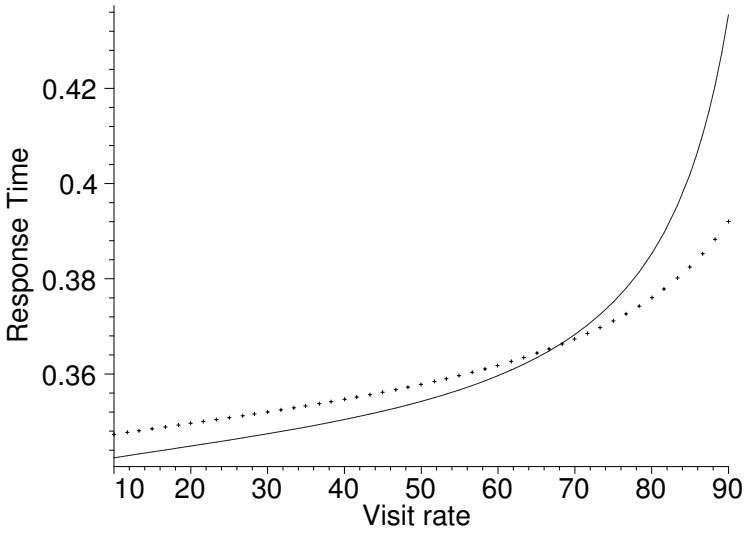

Figure 4: $p = 0.1$, $F = 5000$ bytes,$\Lambda = 150$, K=100

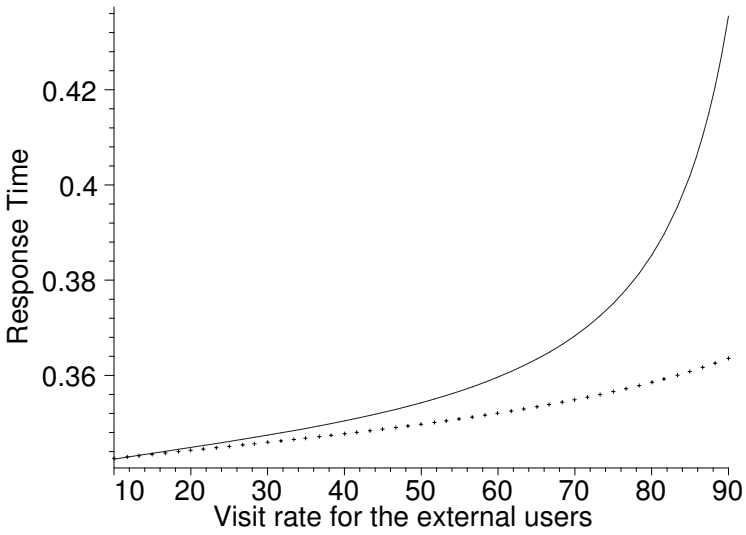

Figure 5: $p = 0.25$, $F = 5000$ bytes,$\Lambda = 150$, K=100

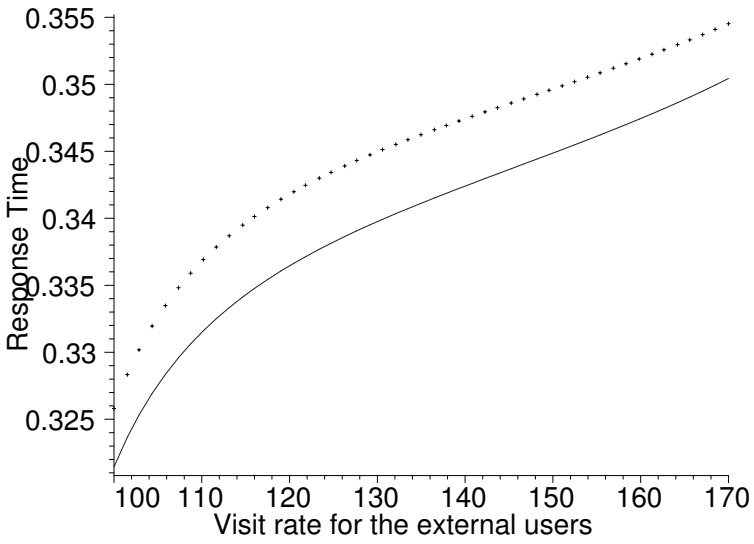

Figure 6: $\lambda = 20$, $p = 0.1$, $F = 5000$ bytes, K=100

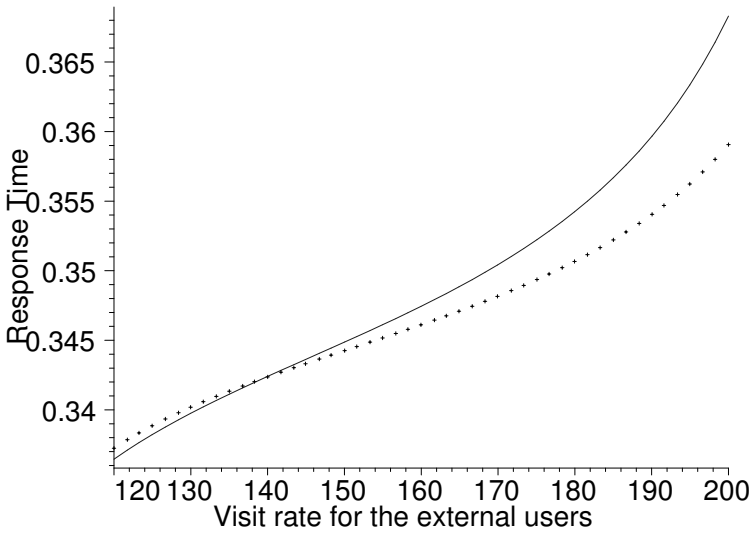

Figure 7: $\lambda = 20$, $p = 0.25$, $F = 5000$ bytes, K=100

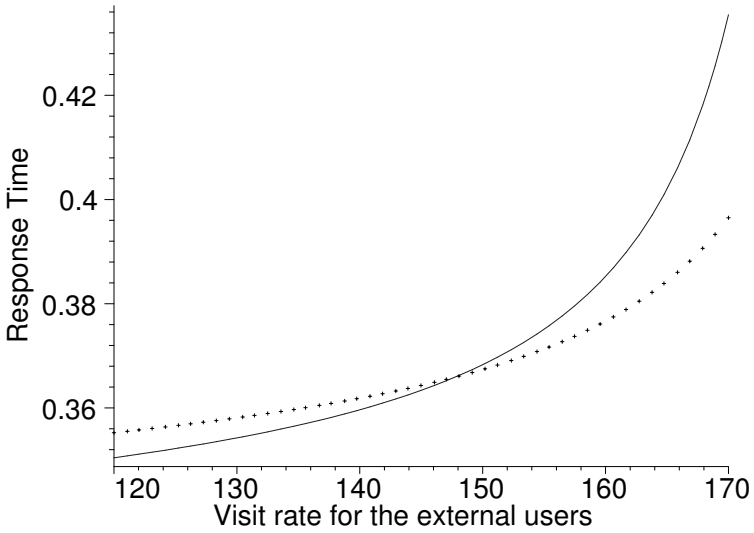

Figure 8: $\lambda = 70$, $p = 0.1$, $F = 5000$ bytes, K=100

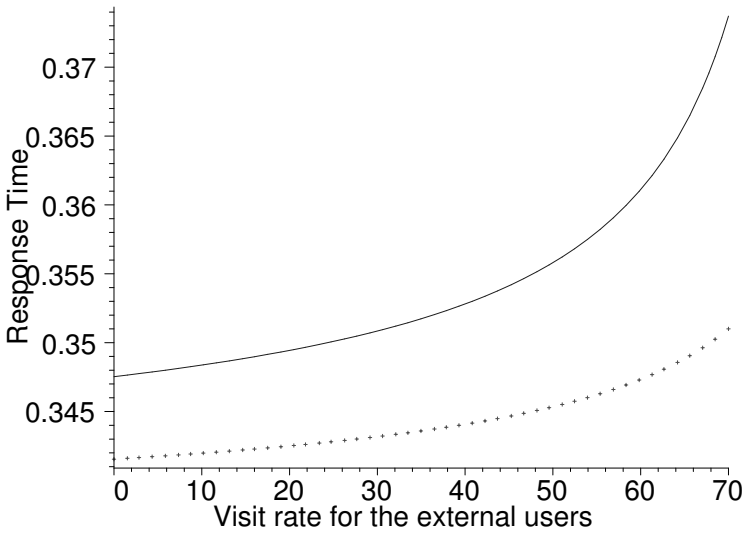

Figure 9: $\lambda = 20$, $p = 0.4$, $F = 5000$ bytes, K=100

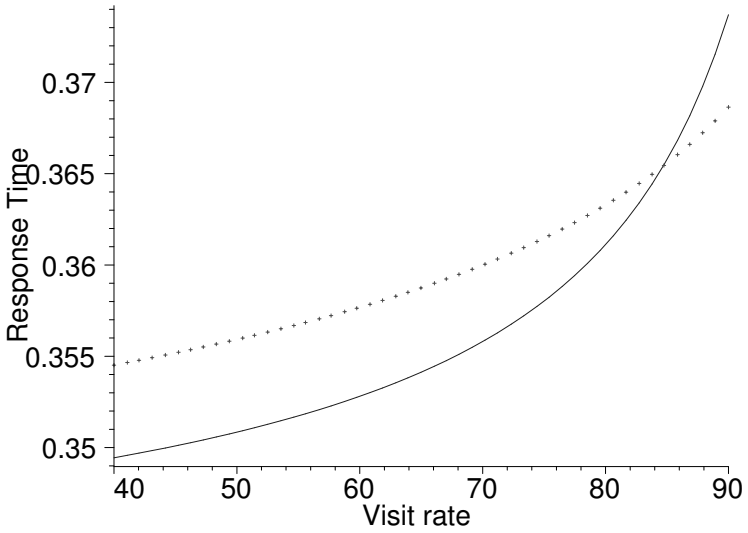

Figure 10: $p = 0.1$, $F = 5000$ bytes, $\Lambda = 0$

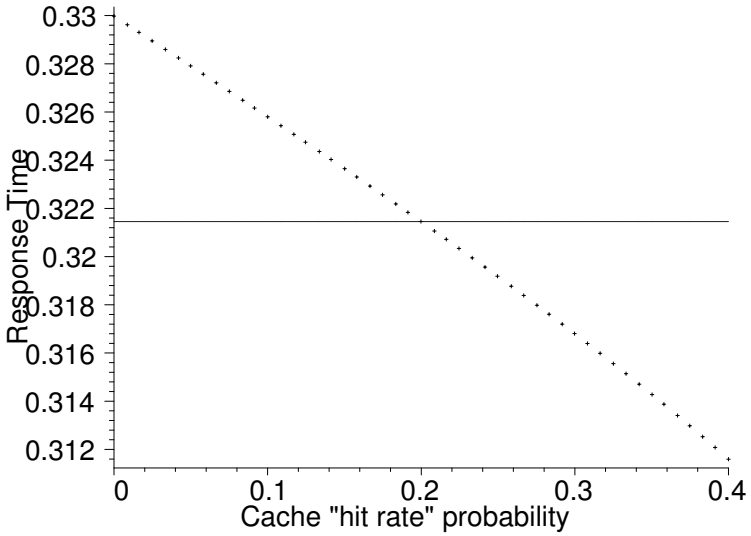

Figure 11: $\lambda = 20$, $\Lambda = 100$, $F = 5000$ bytes, K=100

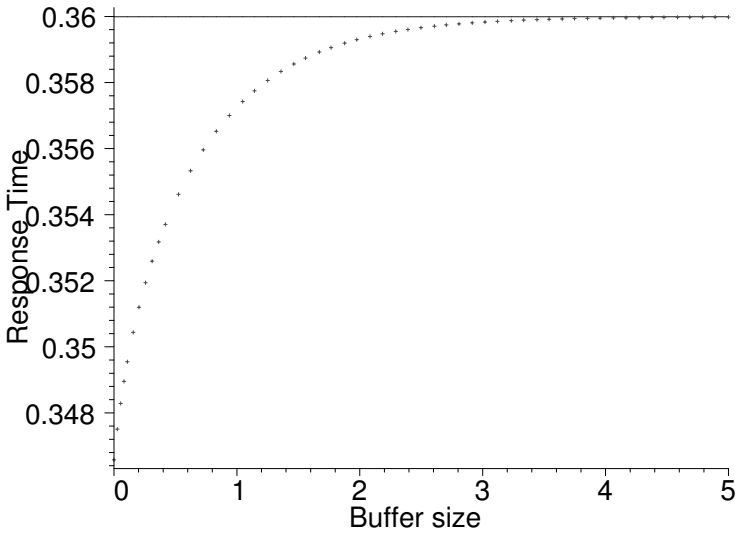

Figure 12: $p = 0.1, \lambda = 70, \Lambda = 0, F = 5000$ bytes

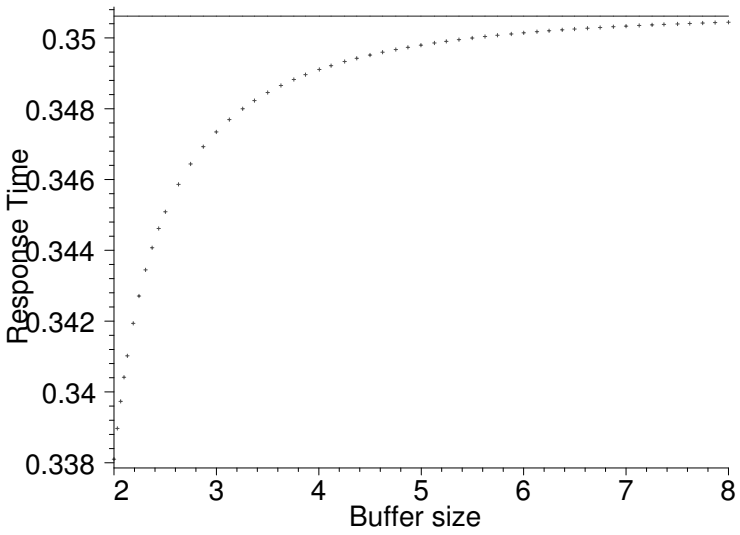

Figure 13: $p = 0.1, \lambda = 70, \Lambda = 100, F = 5000$ bytes

λ: arrival rate from the PCS
Λ: visit rates for the external users
F: average file size (in byte)
p: cache hit rate probability
B_{xc}: PCS output buffer (in byte)
I_{xc}: lookup time of the PCS (in second)
Y_{xc}: static server time of the PCS (in second)
R_{xc}: dynamic server time of the PCS (in byte/second)
N_c: client network bandwidth (in bit/second)
B_s: Web output buffer (in byte)
I_s: lookup time of the Web server (in second)
Y_s: static server time of the Web server (in second)
R_s: dynamic server time of the Web server (in byte/second)
N_s: server network bandwidth (in bit/second)
K: the buffer size of the Web server (in requests)

Table 1: Notations

4 Conclusions

We modified the queueing network model of Bose and Cheng [1] to a more realistic case when external arrivals are allowed to the remote web server and the web server has limited buffer. To examine this model we conducted numerical experiments adapted to realistic parameters. We noticed that, when the arrival rate of requests increases, then the response times increase as well regardless of the existence of a PCS. But in contrast with [1] when external visits are allowed to the remote web server, the PCS was beneficial with a low traffic and a low cache hit rate. When we used a high visit rate with a high cash hit rate probability, then the response time gap was more significant between the cases with and without a PCS.
To compare the two models we examined the effect of the visit rate for the external users. With low external arrival rate installing a PCS resulted higher response times. Increasing the visit rate for the external users, the difference between response time with and without a PCS was smaller and smaller until this difference vanished and the existence of a PCS resulted lower response times.
Examining numerical results it was clear that allowing external arrivals and limited buffer a more realistic model was obtained.

References

1. BOSE, I. , CHENG, H.K., Performance models of a firms proxy cache server. *Decision Support Systems and Electronic Commerce.*, **29** (2000), 45–57.

2. CACHEFLOW INC., 1999. CacheFlow White Papers. Available from ¡http://cacheflow.com/technology/wp¿.
3. LASHINSKY, A., Suddenly cache is king the world of net stocks. *Fortune.*, (1999), 370–372.
4. MENASCE, D.A. , ALMEIDA, V.A.F., Capacity Planning for Web Performance: Metric, Models, and Methods. *Prentice Hall.*, (1998)
5. RUBENSTEIN, R. , HERSCH, H.M., LEDGARD, H.F., The Human Factor: Designing Computer Systems for People. *Digital Press, Burlington, MA.*, (1984)
6. ZHAO, J.L. , KUMAR, A., Data management issues for large scale, distributed workflow systems on the internet. *ACM SIGMIS Data Base.*, **29** (4), 22–32.
7. L.P. SLOTHOUBER, A model of Web server performance. *5th International World Wide Web Conference, Paris, Farnce.*, (1996)