社団法人 情報処理学会　研究報告
IPSJ SIG Technical Report

2004－MPS－50　(2)
2004／6／22

逐次アクセス資源の応答時間への影響の待ち行列網による解析に関する一考察

木　下　俊　之 †

ファイルの一致性を保つために，ひとつのジョブがアクセスしている間は他のジョブからのアクセスが禁止される場合がある．この逐次アクセス資源へのアクセスの衝突は，システム性能に大きな影響を及ぼす．本論文では，アクセスの衝突による資源待ちがジョブの応答時間に及ぼす影響を調べた．そのため資源と資源待ち行列を付加したセントラルサーバモデルを考え，近似のマルコフ連鎖を導入して，ジョブが資源待ち行列に入った回数で条件付けられた条件付平均応答時間を計算した．その結果，アクセスの衝突による資源待ちが，ジョブの応答時間に大きな影響を及ぼすことを確認できた．

Queuing Network Model for Analyzing Influence of Exclusively Used Resource on Performance of Computer Systems

Toshiyuki Kinoshita †

Job conflicts occur when accessing exclusively used resources in a computer system. In this paper, a queuing network model is introduced to analyze the influence conflicts have on the performance of computer systems. It is formed from an ordinary central server model by adding a resource and a resource waiting queue. To analyze how the number of visits of a job to the resource waiting queue affects its response time, the conditional expectation of the response time for the job conditioned by the number of visits to the resource waiting queue is approximately calculate by introducing an approximate Markov chain. The numerical results reveal that the number of conflicts significantly affects the response time of the job.

1. Introduction

Job conflicts occur when accessing exclusively used resources in a computer system. These conflicts affect the performance of the system significantly. In this paper, I proposed a queuing network model with resource requirements and investigate how these conflicts affected the response time of a job. I approximately calculated the conditional expectation of the response time for the job conditioned by the number of conflicts the job encountered by introducing an approximate Markov chain. The numerical results revealed that the number of conflicts significantly affected the response time of a job.

† （株）日立製作所システム開発研究所
Systems Development Laboratory, Hitachi, Ltd.

2. Model Description

2.1 Central server model with resource requirements

I modified an ordinary central server model to have an *exclusively used resource* and *resource waiting queue* (RWQ) (see Figure 1). It includes single CPU node (node number m=0, service rate is μ_0), multiple I/O nodes ($m = 1, 2, \ldots, M$, service rate is μ_m) and a RWQ (referred as $m = M + 1$). There are fixed number N of jobs (job number $n = 1, 2, \ldots, N$). The jobs are scheduled with the FCFS principle in the RWQ as well as at all nodes. A job may request or release the resource at the end of service at the CPU node. If a job requested the resource while occupied, the job joins the RWQ and waits for release of the resource. When the job needs the resource (i.e., requires or

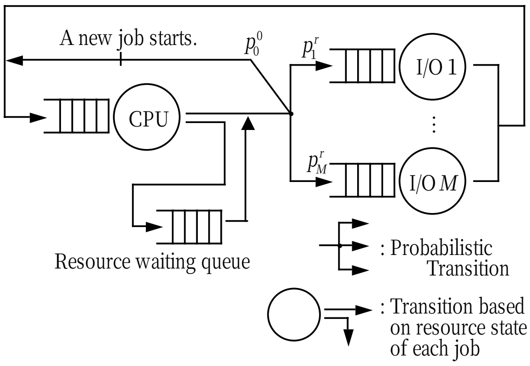

Figure 1. Central server model with resource requirements.

acquires the resource), *resource state* $r = 1$ and when it does not need the resource, $r = 0$. Let $p^{r_1 r_2}$ $(r_1, r_2 = 0, 1)$ denote the transition probability from resource state r_1 to r_2. After leaving CPU node, the job whose resource state is r selects the destination node m at probability p_m^r.

When a job makes a direct transition CPU $\rightarrow$ CPU, the job is regarded as having terminated its life and a new job starts afresh. Hence, the *response time* of a job is defined as its lifetime. Since a job cannot complete while occupying the resource, $p_0^1 = 0$ is assumed.

2.2 State of system

Let N_m be the number of jobs in node m, and N_m^r be the number of jobs whose resource state is r in node m $(r = 0, 1; M = 0, 1, \ldots, M, M+1)$.

The *state of the system* is represented by vector

$$\delta = (k_0^1 \cdots k_0^{N_0};\ k_1^1 \cdots k_1^{N_1};\ \cdots\ ;\ k_M^1 \cdots k_M^{N_M};\ k_{M+1}^1 \cdots k_{M+1}^{N_{M+1}})\,,$$

where variable k_m^q stands for resource state r, which indicates that the q-th position at node m is occupied by a job whose resource state is r $(m = 0, 1, 2, \ldots, M, M+1;\ \ q = 1, 2, \ldots, N_m;\ r = 0, 1)$. Hence, there is at most one value 1 in $k_0^1 \cdots k_0^{N_0};\ k_1^1 \cdots k_1^{N_1};\ \cdots\ ;\ k_M^1 \cdots k_M^{N_M}$, and all $k_{M+1}^1 \cdots k_{M+1}^{N_{M+1}}$ are value 1s since all jobs in the RWQ have resource state 1.

The stochastic behavior of state transition is described by a Markov chain $\{\delta(t)\}$ on set Δ of all possible δs.

2.3 Performance measures in steady-state

Almost all important performance measures for the system can be calculated from steady-state probabilities $P(\delta)$ of the Markov chain $\{\delta(t)\}$.

Busy ratio ρ_m^r and throughput λ_m^r at the CPU node and I/O nodes by jobs whose resource state is r is given by

$$\left.\begin{aligned} \rho_m^r &= \sum_{\substack{\delta \in \Delta \text{ s.t.} \\ N_m^0 > 0 \ \&\ k_m^1 = r}} P(\delta)\,, \\ \lambda_m^r &= \mu_m\, \rho_m^r\,, \end{aligned}\right. \qquad \begin{pmatrix} r = 0, 1; \\ m = 0, 1, 2, \ldots, M \end{pmatrix}$$

respectively.

Throughput λ of the system, which is the frequency of occurring job completion, is given by $\lambda = p_0^0\, \lambda_0^0$. Moreover, throughput λ_{M+1} of the RWQ is given by

$$\lambda_{M+1} = \mu_0\, p^{01} \cdot \sum_{\substack{\delta \in \Delta \text{ s.t.} \\ N_0^0 > 0,\ k_0^1 = 0\ \& \\ \text{there exists one 1 in } \delta}} P(\delta) = \lambda_0^0\, p^{01}\, q\,,$$

where q is the conditional probability for a job going to the RWQ when the job requests the resource. Therefore, q is calculated by the equation

$$q = \sum_{\substack{\delta \in \Delta \text{ s.t.} \\ N_0^0 > 0,\ k_0^1 = 0\ \& \\ \text{there exists one 1 in } \delta}} P(\delta) \Bigg/ \sum_{\substack{\delta \in \Delta \text{ s.t.} \\ N_0^0 > 0\ \&\ k_0^1 = 0}} P(\delta)\,. \qquad (1)$$

The mean number of jobs whose resource state is r in node m is given by

$$L_m^r = \sum_{\delta \in \Delta} N_m^r P(\delta) \quad (m = 0, 1, 2, \ldots, M, M+1).$$

Hence, the mean residence time T_m^r of a job whose resource state is r in node m per visit and the mean response time T of a job are given by

$$T_m^r = \frac{L_m^r}{\lambda_m^r} \quad \text{and} \quad T = \frac{N}{\lambda}\,. \qquad (2)$$

3. Response time and number of visits to resource waiting queue

In this section, let us investigate how the number of visits of a job to RWQ u_{M+1} affects response time t of the job. I introduce an approximate Markov chain describing the job state transitions of a tagged job and approximately calculate the conditional expectation $E(t \,|\, u_{M+1} = k)$ by assuming

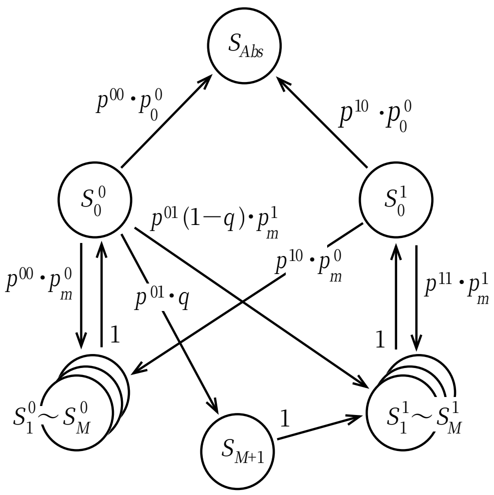

Figure 2. State transition of Markov chain $\{\gamma^*(n)\}$

that the mean residence time at each node will not depend on the history of job state transitions.

My approximate Markov chain $\{\gamma^*(n)\}$ is an absorbing Markov chain which has states as follows (see Figure 2).

S_{Abs} : absorbing state indicating completion of tagged job,

S_m^r : state of tagged job whose resource state is r at node m ($r = 0, 1$; $m = 0, 1, 2, \ldots, M$),

S_{M+1} : state of tagged job at RWQ.

I have assumed this transition probability from S_0^0 to S_{M+1} of the tagged job is independent of the history and equal to $p^{01}q$, where q is the conditional probability of a job going to the RWQ introduced in Equation 1.

Let u_m^{0*}, $u_m^{1*}(m = 0, 1, 2, \ldots, M)$ and u_{M+1}^* be the numbers of visits to states S_m^0, S_m^1 and S_{M+1} in the Markov chain $\{\gamma^*(n)\}$ respectively, and define

$$t^* = \sum_{m=0}^{M} T_m^0\, u_m^{0*} + \sum_{m=0}^{M} T_m^1\, u_m^{1*} + T_{M+1}\, u_{M+1}^*,$$

where T_m^0, T_m^1 and T_{M+1} are introduced in Equation 2. By using the conditional probability generating function of u_{M+1}^* conditioned so that the chain starts from state S_m^0, i.e.,

$$g_m^0(s) = \sum_{k=0}^{\infty} P(u_{M+1}^* = k \mid \gamma^*(0) = S_m^0)\, s^k \qquad (m = 0, 1, 2, \ldots, M),$$

and similarly defined $g_m^1(s)$ $(m = 0, 1, 2, \ldots, M)$, $g_{M+1}(s)$ and $g_{Abs}(s)$ for each starting state, we have

$$P(u_{M+1}^* = k \mid \gamma^*(0) = S_0^0) = \begin{cases} \dfrac{p_0^0(1-p^{01}q)}{p_0^0 + p^{01}q(1-p_0^0)} & \text{for } k = 0, \\[2ex] \dfrac{p_0^0}{(1-p_0^0)\{p_0^0 + p^{01}q(1-p_0^0)\}} \times \left\{ \dfrac{p^{01}q(1-p_0^0)}{p_0^0 + p^{01}q(1-p_0^0)} \right\}^k & \text{for } k = 1, 2, \ldots . \end{cases} \tag{3}$$

This conditional probability can serve as an approximation of $P(u_{M+1} = k)$.

By using the generating function $f_m^0(s)$ for the conditional expectation of t^* as

$$f_m^0(s) = \sum_{k=0}^{\infty} E(t^* \mid u_{M+1}^* = k, \gamma^*(0) = S_m^0) \times P(u_{M+1}^* = k \mid \gamma^*(0) = S_m^0)\, s^k \qquad (m = 0, 1, 2, \ldots, M),$$

and similarly defined $f_m^1(s)$ $(m = 0, 1, 2, \ldots, M)$, $f_{M+1}(s)$ and $f_{Abs}(s)$ for each starting state, we have

$$E(t^* \mid u_{M+1}^* = k, \gamma^*(0) = S_0^0) = \begin{cases} \dfrac{1}{p_0^0 + p^{01}q(1-p_0^0)} \left\{ T_0^0 + T_{IO}^0 + \dfrac{p^{01}(1-q)}{p^{10}}(T_0^1 + T_{IO}^1) \right\} - T_{IO}^0 \\ \qquad \text{for } k = 0, \\[2ex] \dfrac{1}{p_0^0 + p^{01}q(1-p_0^0)} \left\{ T_0^0 + T_{IO}^0 + \dfrac{p_0^0 + p^{01}(1-p_0^0)}{p^{10}}(T_0^1 + T_{IO}^1) \right\} \cdot (k+1) \\ \quad - \left\{ T_0^0 + 2 \cdot T_{IO}^0 + \dfrac{1}{p^{10}}(T_0^1 + T_{IO}^1) \right\} \\ \quad + T_{M+1}\, k \qquad \text{for } k = 1, 2, \ldots , \end{cases} \tag{4}$$

where $T_{IO}^0 = \dfrac{1}{1-p_0^0} \displaystyle\sum_{m=1}^{M} T_m^0\, p_m^0$, $T_{IO}^1 = \displaystyle\sum_{m=1}^{M} T_m^1\, p_m^1$. This conditional expectation can serve as an approximation of $E(t^* \mid u_{M+1}^* = k)$.

Table 1. Influence of visits to resource waiting queue

(a) $N = 6$

	$P(u_{M+1} = k)$		$E(t\|u_{M+1} = k)$	
k	Appox.	Sim.	Approx.	Sim.
0	0.727	0.726	9.78	10.05
1	0.217	0.215	35.72	34.80
2	0.045	0.046	65.26	63.69
3	0.0094	0.0101	94.81	92.42
4	0.0019	0.0022	124.35	121.75
5	0.0004	0.0005	153.90	149.02

$$P(u_{M+1} = k) \sim 1.042 \times 0.208^k$$
$$E(t|u_{M+1} = k) \sim 6.170 + 29.546 \cdot k$$
for $k = 1, 2, \ldots$

(b) $N = 12$

	$P(u_{M+1} = k)$		$E(t\|u_{M+1} = k)$	
k	Appox.	Sim.	Approx.	Sim.
0	0.684	0.682	10.01	10.10
1	0.240	0.242	73.50	72.83
2	0.058	0.058	140.88	139.93
3	0.014	0.014	208.27	207.64
4	0.0033	0.0034	275.65	275.35
5	0.0008	0.0008	343.04	342.96

$$P(u_{M+1} = k) \sim 0.999 \times 0.240^k$$
$$E(t|u_{M+1} = k) \sim 6.115 + 67.384 \cdot k$$
for $k = 1, 2, \ldots$

4. Numerical experiments

4.1 Parameters of numerical experiments

To determine the effects of resource requirements, I numerically analyzed the model and calculated its performance measures for several sets of parameter values as follows:

(1) $N = 6$, 12 (2) $M = 2$

(3) $\mu_0 = 2.0$ (4) $\mu_1 = \mu_2 = 1.0$

(5) $p^{01} = 0.1$, $p^{00} = 1 - p^{01} = 0.9$

(6) $p^{10} = 0.5$, $p^{11} = 1 - p^{10} = 0.5$

Probability p^{11} is concerned with the number of circulating the network with the resource in succession. $p^{11} = 0.5$ implies that a job will release the resource after 2 circulations on average.

(7) $v_0 = 5$

(8) $p_0^0 = \dfrac{1}{v_0}\left(1 + \dfrac{p^{01}}{p^{10}}\right) = 0.24,$

since $p^{01} = 0.1$, $p^{10} = 0.5$, and $v_0 = 5$.

(9) $p_m^1 = \begin{cases} 0.4 & \text{for } m = 1, \\ 0.6 & \text{for } m = 2. \end{cases}$

(10) $p_m^0 = \dfrac{1 - p_0^0}{2} \quad \text{for } m = 1, 2\,.$

4.2 Numerical results

Table 1 lists numerical results for $P(u_{M+1} = k)$ approximately calculated as $P(u^*_{M+1} = k \,|\, \gamma^*(0) = S_0^0)$ based on Equation 3, $E(t \,|\, u_{M+1} = k)$ approximately calculated as $E(t^* \,|\, u^*_{M+1} = k, \gamma^*(0) = S_0^0)$ based on Equations 4, and their corresponding simulation results for $p^{01} = 0.1$ and $N = 6$ and 12. We can see that these approximate values are very close to the simulation results and this approximate model is quite accurate. These results also indicate that the mean response time of a job is very long once it joins the RWQ, since the mean residence time of a job in the queue becomes very long when p^{01} or N is larger.

5. Conclusion

I introduced a central server model with resource requirements. By using an approximate Markov chain, I approximately represented the conditional expectation $E(t \,|\, u_{M+1} = k)$ of the response time for the job and clarified that the mean response time for the job became very long once the job had visited the resource waiting queue when the resource became a bottleneck.

参 考 文 献

1) Kinoshita, T., Takahashi, Y., A Queuing Network Modeling and Performance Evaluation Method for Computer Systems with Resource Requirement (In Japanese), *Trans. of IEICE of Japan*, **82-D-1** (6), pp. 701–710 (1999).

2) Kinoshita, T., Takahashi, Y., An Approximate Method by Queuing Network Modeling for Performance Evaluation Method for Computer Systems with Exclusively-Used Resources (In Japanese), *Trans. of Information Processing Soc. of Japan*, **42** (SIG 14 TOM 5), pp.1–13 (2001).