Repeat Task Sending Model and Performance Analysis of Conversational Mode Multiple Computer System

Chugo FUJIHASHI* and Mitsuyoshi KITAMURA**

ABSTRACT

For more appropriate analysis of a conversational mode multiple computer system performance influenced by stochastic arrival, typing and demanding properties of users, a repeat task sending model is presented. In the model, a remain probability R characterizing the repeat task sending property of a user is introduced as probability that the user remains at terminal after he receiued his served tasp. The equations to the probabilities for queue sizes are derived for the analysis, and the probabilities satisfying the equations are obtained in simple multiplication form. Various perfomrance measures based on the probabilities, including average response time and average holding time of terminal, are given. Finally properties of several performance measures are numerically computed and shown in figures.

1. INTRODUCTION

Distributed and parallel computing systems are regarded as the centerpiece of advanced future computer systems[1], and many theoretical and experimental researches on software and architecture have been reported. There are many difficult problems which should be overcome, the performance evaluation is one of most important issues among them.

For the performance evaluation of computer system, an idea of time is important. Enpecially response time is important for a real time distributed system, and the analysis on the response time by submodel and decomposition technique had been reported[2]. In response time, serving delay has key role, and the delay mainly depends on capacities of computers and scheduler in the distributed computing system. Then the optimization technologies for the static[3] and the dynamic[4], [5] schedulers are addressed. Performance of complex computer network are evaluated by not only response time but also other total measures. The load balancing for homogeneous[6] and heterogeneous[7] networks and the traffic characterization of networks[8] are researched.

For the above problems it is necessary to develop various approaching methods to give basic materials for designs and analyses. Experimental approaches are important for evaluations of real systems, and have been used

*The Department of Electronic Engineering
**Research and Education Center of Software Engineering, Tokyo Institute of Polytechnics
Received Sep. 10, 1996

for the multiprocessor system[9] and the work-station traffic measurement[10]. Generally simulation analysen are also powerful for the performance evaluations[11], however, the analyses must spend long computing time for large system. Theoretical approaches, the theory based on queueing, can give direct ways for the problems, and the B. C. M. P. theorem[12] is known at present day as the unsolved important problems. Therefore various theoretical approaches for the problems should be developed.

Among the problems stochastic properties of users are important in real time distributed computer system. Indeed response time in real time system is mainly decided by interactions of user prokerties and service capacities of computers. The user properties, however, have not been grasped in detaiq, and the theory which can clearly describe the user properties in syster performance is not established yet. The close system model in the theory[12] only can deshribe simple pmoperty of users. Then a generalized and modified queueing model is necessary for appropriate analyses of user role in the performance of computer systems.

In this paper a repeat tasp sending and annihilation model is presented for analysis of user influences in system performance of a conversational mode real time multiple computer network. The user in the model comes from external area to the system and holds one of free terminals and begin his work. The user who could hold a terminal sends a task to the computers, and annihilates the served task returned to the user, asd the cycle of the sending and annihilation is repeated until finish of his work.

Section 2 describes a configuration diagram including computers, terminals and internal connection, and a queueing model corresponding to the configuration diagram. In section 3 the equations to probabilities ftor queue sizes are derived and the solution under equilibrium state condition is given. Section 4 offers several performance measures, and section 5 gives the results of numerical computations of the performance measures.

2. SYSTEM and QUEUEING MODEL

The real time distributed computing network system cosidered in this paper is shown in Fig. 1. The system is basically constituted by two main components of the multiple terminal (the cluster of N terminals) and the multiple computer (the cluster of S computers), and operates in conversational mode. In the system, each of the terminals is connected to each of the computers by connection networks. Therefore a demand for computation (a task) sent from one of the terminals can be accepted by any of the computers in cluster.

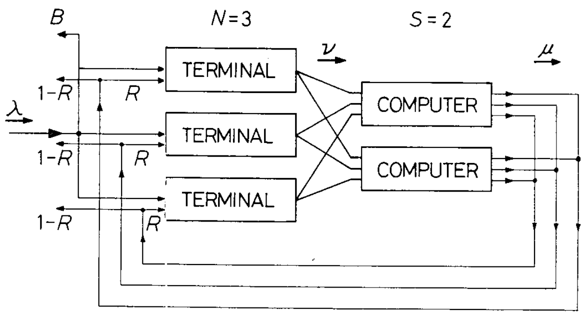

Fig. 1 Conversational mode multiterminal and multicomputer system.

A user arriving at the cluster of terminals selects and holds one of free terminal. and begins his work. Generally the work consists of services for tasps at computers and preparations for the tasks. The preparation means arrangement including user thinking, reading and typing to send the tasps, and the arranged tasks are sent to the cluster of computers, and each of the tasks is served at one of the computers. The tasp offered a service by the computer is returned back to the user who sent the task. The returned task is annihilated immediately after the returning. Therefore, the tasps are sent and annihilated on terminal by users, and are not entered from external area.

In a conversational mode computer system, a user may repeatedly send many tasks, and remains until finish of his work including services for tasks. Therefore, it is important for practical engineering to set up a model which gives possibility for analyses of the performance affected by the repeat sending of tasks and service capacities of the computers.

For the analysis of influences of the repeat task sending in system performance, a remain probability R is introduced. The remain probability R is a parameter which is to represent stochastic situation that a user remains for his further work immediately after he has received his served task. The departure probability of a user after receiving of his semved task becomes $1-R$. The assumption, that the selection of remaining or departure is done immediately after the receiving ta the served task and a fixed value is simply given to the probability R, may not be exact for real system. It is regarded, cowever, as reasonable for analyses of principal chasacteristics of system performance with the repeat task sending.

In the case where tasks are repeatedly sent and served, there are two types of service operations for a new sent task in relation to an old task which had been sent by the same user who have sent the new task. In the first type, all the computers serve independently, and a new task and an old task sent by the same user can not be served at different computers. In the second type, all the computers serve cooperatively and a new task and an old task by same user can be served at arbitrary different computers. Then the former can be called an independent service system, and the latter can be called a cooperative service system.

The difference between the former and the latter is important for the performance analysis of distributed computing systems. The latter is regarded as a typical model for future's parallel and distributed multiple computer systems. The latter system is considered in this paper, though the system is not yet conventional at present day.

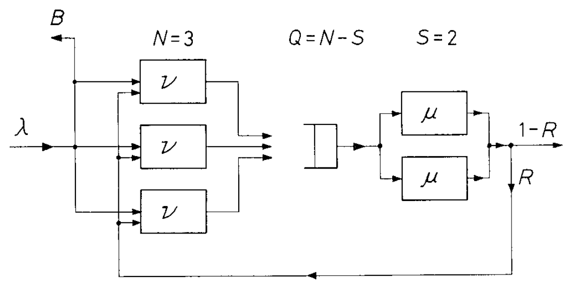

Fig. 2 Queueing model for the conversational mode multicomputer system.

A queueing model for the system in drawn in Fig. 2. The system has a single queue in front of the cluster of computers, and capacity for waiting is given by $N-S$. It is important that queues permitted or not for the cluster of terminals, however, such the queues in front of terminals are not usually formed in almost conversational mode computer systems since a holding time of terminals by a user in real system is customary far longer than the patientable waiting time for a user. Then a model with no queue in front of the cluster of terminals is assumed. The system is regarded as noe of loss systems, and loss probability for the system is represented by B.

3. ANALYSIS

Before analyzing the queueing model, several mathematical symbols necessary for the following theoretical treatment are introduced. Let λ

be arrival rate of a user to the system, ν be sending rate of a task for computation on a terminal by a user, and μ be service rate for the task at a computer. Here sending rate of a task for computation means equivalently termination rate of preparation for the task on a terminal. For simplicity the term of sending rate only is used in this paper. In the model all stochastic properties of arrival, sending and service are assumed to be described by Poisson processes (Appendix 1).

3.1 SOLUTION

In this chapter the equations for analyses of the above queueing system are given, and derivation of the solution is described. Because the characteristics under the equilibrium state condition are sufficient for practical engineering, the following analyses are proceeded under assumption of the equilibrium state condition.

Let (n, m) be the state that the number of users who are preparing tasks in n and the number of tasks which are waiting or in service is m, and $p(n, m)$ be the probability that the state (n, m) is find out arbitrary time. The equations to the probabilities for $n\geq 0$ and $m\geq 0$ and $n+m\leq N$ are derived

$$\left\{\underset{\#1}{\underline{(1-\delta_{n+m,N})\lambda}}+\underset{\#2}{\underline{n\nu}}+\underset{\#3}{\underline{(1-R)g_m\mu+Rg_m\mu}}\right\}p(n, m)$$
$$=\underset{\#2}{\underline{\lambda p(n-1, m)}}+\underset{\#3}{\underline{(n+1)\nu p(n+1, m-1)}}+\underset{\#1}{\underline{(1-R)g_{m+1}\mu p(n, m+1)}}+\underset{\#2}{\underline{Rg_{m+1}\mu p(n-1, m+1)}} \quad (1)$$

where

λ : A service rate of a computer
ν : An arrival rate of a user
μ : A sending rate at a terminal by a user and

$$p(-1, m)=p(n, -1)=0$$
$$p(n, m)=0 \quad (n+m>N)$$
$$\delta_{r,N}=\begin{cases}1 & (r=N)\\ 0 & (r\neq N)\end{cases}$$
$$g_m=\begin{cases}m & (m\leq S)\\ S & (m>S)\end{cases} \quad (2)$$

Each of the equations for the state (n, m) is derived from the epuilibrium state condition for total of the transitions between the state (n, m) and the all other states, and called the global balance equation[12]. A transition diagram ftm the global balance equation is shown in Fig. 3.(a).

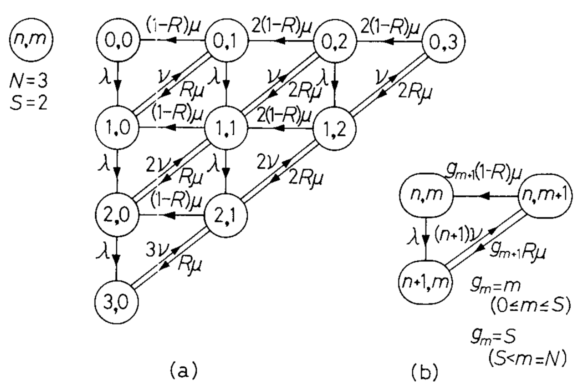

Fig. 3 Transition diagrams for the global valance equation (a) and for the independent valance equations (b).

Similarly to many other global balance equations for practical complex problems, direct derivation of a solution for Eq.(1) generally may not be simple. For such the equations it is known that the techngque of independent balance equations is sometimes efficient, though the technique is not applicable for all cases. Applying the technique to Eq.(1), the independent balance equations are obtained from each underlined corresponding terms of #1, #2, and #3 in Eq.(1). The each equatitns from terms of #1, #2 and #3 become

$$\lambda p(n, m)=(1-R)g_{m+1}\mu p(n, m+1) \quad (3)$$
$$n\nu p(n, m)=\lambda p(n-1, m)+Rg_{m+1}\mu p(n-1, m+1) \quad (4)$$

$$g_m \mu p(n, m) = (n+1)\nu p(n+1, m-1) \qquad (5)$$

A transition diagram for the independent balance equations is shown in Fig. 3(b).

The solution is derived as follows. From (3) next recursion relation is obtained.

$$p(n, m+1) = \frac{\lambda}{(1-R)g_{m+1}\mu} p(n, m) \qquad (6)$$

Substitute Eq.(6) into the small changed left hand side $g_{m+1}\mu p(n, m+1)$ in Eq.(5), another recursion relation is obtained.

$$p(n+1, m) = \frac{\lambda}{(1-R)(n+1)\nu} p(n, m) \qquad (7)$$

From the above recursion relations of Eq.(6) and (7) the solution for $p(n, m)$ finally becomes $0 \le m \le S$

$$p(n, m) = \frac{1}{n!m!}\left\{\frac{\lambda}{(1-R)\nu}\right\}^n \left\{\frac{\lambda}{(1-R)\mu}\right\}^m p(0, 0) \qquad (8a)$$

$S \le m \le N$

$$p(n, m) = \frac{1}{n!S!S^{m-S}}\left\{\frac{\lambda}{(1-R)\nu}\right\}^n \left\{\frac{\lambda}{(1-R)\mu}\right\}^m p(0, 0) \qquad (8b)$$

where

$$p(0, 0) = 1\Bigg/\left[\sum_{m=0}^{S}\sum_{n=0}^{N-m}\frac{1}{n!m!}\left\{\frac{\lambda}{(1-R)\nu}\right\}^n\left\{\frac{\lambda}{(1-R)\mu}\right\}^m + \sum_{m=S+1}^{N}\sum_{n=0}^{N-m}\frac{1}{n!S!S^{m-S}}\left\{\frac{\lambda}{(1-R)\nu}\right\}^n\left\{\frac{\lambda}{(1-R)\mu}\right\}^m\right] \qquad (8c)$$

Eq.(8c) is obtained by well known method based on the formula in probability theory.

$$\sum_{m=0}^{N}\sum_{n=0}^{N-m} p(n, m) = 1 \qquad (9)$$

Eqs.(8a), (8b) and (8c) satisfy the independent balance equations, and then satisfy the global balance equation. Hence EQs.(8a), (8b) and (8c) give the solution for Eq.(1).

3.2 MEANING of REMAIN PROBABILITY

In chapter 2, R was introduced as the probability that a user remains at terminal immediately after he has received his served task. To make the meaning of R clear in the solution given by Eqs.(8a), (8b) and (8c), more detail examination on property of R is necessary.

In the explanation in chapter 2, 1-R is the probability that a user completes (or balks) his work and departs from the system immediately after he has received his served task. From the probability 1-R, it is possible to derive the probability $p(r)$ that the user just departs from the system immediately after he received his n th served task. The probability is obtained as product of the probability R^{n-1} that the user remained after he had received $(n-1)$th served task (or until he receives nth served task) and the probability $1-R$ that the user just departs after he has received a served task.

$$p(0) = 0 \qquad (10a)$$
$$p(r) = R^{r-1}(1-R) \qquad (10b)$$

From the probability Eq(10a) and (10b), the average number of served tasks which a user receives before his departure can be obtainde. Let M be the average number, M is obtained by the following calculations.

$$M = \sum_{r=0}^{\infty} rp(r) = (1-R)\sum_{r=0}^{\infty} rR^{r-1} = (1-R)\frac{d}{dR}\sum_{r=0}^{\infty} R^r = (1-R)\frac{d}{dR}\frac{1}{(1-R)} = \frac{1}{1-R} \qquad (11)$$

The average number M is equal to the number of tasks meanly served in a work, and is equivalent to an average repeat number of tasks sent in the work. Because the repeat number of task sendinb is important for performance measures, and if the solution is rewritten by the average repeat number M, the solution becomes

$0 \leq m \leq S$

$$p(n, m)=\frac{1}{n!m!}\left\{\frac{\lambda}{\nu}M\right\}^n\left\{\frac{\lambda}{\mu}M\right\}^m p(0, 0) \tag{12a}$$

$S < m \leq N$

$$p(n, m)=\frac{1}{n!S!S^{m-S}}\left\{\frac{\lambda}{\mu}M\right\}^n\left\{\frac{\lambda}{\mu}M\right\}^m p(0, 0) \tag{12b}$$

where

$$p(0, 0)=1\Big/\left[\sum_{m=0}^{S}\sum_{n=0}^{N-m}\frac{1}{n!m!}\left\{\frac{\lambda}{\nu}M\right\}^n\left\{\frac{\lambda}{\mu}M\right\}^m + \sum_{m=S+1}^{N}\sum_{n=0}^{N-m}\frac{1}{n!S!S^{m-S}}\left\{\frac{\lambda}{\nu}M\right\}^n\left\{\frac{\lambda}{\mu}M\right\}^m\right] \tag{12c}$$

The expressions mean that the case where M tasks are meanly sent for a work is equivalent to the case where only one task sent for a work and the arrival rate of user is increased by $M\lambda$.

3.3 ASYMPTOTIC SOLUTION at $\lambda\to\infty$

In the previous discussions, it is assumed that users arrive continuously to the system with random stochastic property, and the analyses are done under the assumption. There are, however, the cases where the users equal to the number of the terminals simultaneously come, and begin their works and complete. Examples of such the practical cases are seen at lessons of computer programming exhises.

In the cases, holdings of all the terminals are simultaneously begun, and each terminal is not released independently in a fixed interval of working time. All the terminals are continuously held in the fived interval and simultaneously released. To obtain the solution to dxpress the characteristics in the case where all the terminals are held, it is assumed that the arrival rate λ is taken to be ∞, because any of terminals is not released in the interval and the state of system is regarded approsimately under equilibrium for an enough long interval. Assuming $\lambda\to\infty$ for Eqs. (8a), (8b) and (8c), both terms of order λ^N in the numerators and in the denominators of equations finally become maximum, and the equatios asymptotically become

$0 \leq n \leq S$

$$p(n, m)=\frac{N!}{(N-m)!m!}\left\{\frac{\nu}{\mu}\right\}^m p(N, 0) \tag{13a}$$

$S \leq m \leq N$

$$p(n, m)=\frac{1}{(N-m)!S!S^{m-S}}\left\{\frac{\nu}{\mu}\right\}^m p(N, 0) \tag{13b}$$

and

$$p(N, 0)=1\Big/\left[\sum_{m=0}^{S}\frac{1}{(N-m)!m}\left\{\frac{\nu}{\mu}\right\}^m -\sum_{m=S+1}^{N}\frac{1}{(N-m)!S!S^{m-S}}\left\{\frac{\nu}{\mu}\right\}^m\right. \tag{13c}$$

The results are not depend on the parameter R, and are equal to the known formulae[13].

4. PERFORMANCE EVALUATION

Using the probabilities given by Eqs.(8a), (8b) and (8c) obtained in the previous chapter, various measures to evaluate the system serxice performance can be calculated. In this section several selected important performance measures are described.

4.1 PERFORMANCE CEASURES

(a) Average number of terminal holdings C

The measure indicates the time averaged number of terminals held by users who are preparing the task for service or waiting the served task from computers. The average number of terminal holdings C is obtained as the averaged value of sum of the number of preparing users n and the number of tasks m which are in service or waiting for services.

$$C=\sum_{m=0}^{N}\sum_{n=0}^{N-m}(n+m)\,p(n, m) \tag{14}$$

(b) Average number of waiting tasks L_W

This is the average number of tasks waiting in queue for computing. The average number of waiting tasks Lw is given by the averaged number of $m-S$ since the number of computers is taken S for the system.

$$L_W=\sum_{m=S+1}^{N}\sum_{n=0}^{N-m}(n-S)\,p(n,m) \tag{15}$$

(c) Average number of tasks in service L_C

This is the measure indicating the average number of tasks in service which is meanly found at arbitrary time. The number of tasks in service at computers is given by m where the value of m is smaller than or equal to S, or given by S where m is larger than S. Then an average of the number is given by

$$L_C=\sum_{m=0}^{S}\sum_{n=0}^{N-m}mp(n,m)+S\sum_{m=S+1}^{N}\sum_{n=0}^{N-m}p(n,m) \tag{16}$$

The measure is equivalent to the time averaged number of operating computers.

(d) Average number of tasks in system L_t

The measure is the average number of total tasks found in the system. Then the average number of tasks in the system is given by sum of the average number of waiting tasks L_W and the average number of tasks in service L_C.

$$L_t=\sum_{m=S+1}^{N}\sum_{n=0}^{N-m}mp(n,m)=L_W+L_C \tag{17}$$

This is equal to the average numberof users who are waiting the returning of served tasks from computers.

(e) Average number of preparing users C_P

Because the number of preparing users is given by n, the average number C_P is obtained

$$C_P=\sum_{m=0}^{N}\sum_{n=0}^{N-m}np(n,m) \tag{18}$$

(f) Total service rate A

The measure means the total termination rate of tasks from the cluster of computers. Then the rate is obtained from multiplication of the average number of tasks in service L_C and the service rate μ of a computer.

$$A=L_C\mu \tag{19}$$

(g) Total sending rate E

The total sending rate E means the average number of tasks sent in a unit time from the cluster of terminals. Then the measure is given by muqtiplication of the average number of preparing users C_P and the sending rate ν by a user.

$$E=C_P\nu=A \tag{20}$$

(h) Average waiting time T_W

The number of waiting tasks meanly found at arbgtrary time in front of the computers is represented by L_W, and the total service rate in the system is represented by A. Then the average waiting time is obtained from the same concept for Little's formala[15] (since the average time interval of terminatitn of the tasks from the cluster of computers in equal to $1/A$ in the system), and is given by

$$T_W=L_W/A=L_W/E \tag{21}$$

The value for the measure T_W is obtained from the values of L_W and A by Eq.(15) and (19).

(i) Average response time T_r

This is regarded as the most important measure to evaluate the system performance. The auerage response time is defined as the total time length which the tasp sent to computers meanly spends in queue and in service. Then the measure is obtained from the measures L_t and A in similar manner for T_W.

$$T_r=T_W+1/\mu=L_t/A \tag{22}$$

(j) User departure rate D

The departure rate of the users immediately after receiving of served tasks is given by multiplication of the departure probability of user $1-R$ and the receiving rate equal to the total service rate A. Then the user departure rate D totally becomes

$$D=(1-R)A \tag{23}$$

(k) Average holding time H

The average holding time is defined as an average of time length in which the user holdi a terminal for his work. The measure is obtained from C and D in the similar manner to T_r or T_W, and is expressed by other measures as follows.

$$\begin{aligned} H &= C/D=(C_F+L_t)/\{(1-R)A\} \\ &= M(C_P/E+L_t/A) \\ &= M(1/\nu+T_r) \end{aligned} \tag{24}$$

The last expression shows that the average holding time is equal to M times of sum of the average sending time interval $1/\nu$ and the average response time T_r.

(j) Efficiency O

This is defined as time averaged proportion of computers in operation state.

$$O=L_C/S \tag{25}$$

(k) Loss probability B

It is assumed that a user, who finds the state that all terminal are held, leaves the system and is cleared out. The finding probability is equal to time averaged probability of the state that all N terminals are held by users for preparing or waiting tasks.

$$B=\sum_{m=0}^{N} p(N-m, m) \tag{26}$$

4.2 NUMERICAL COMPUTATION

To evaluate the service performance of the system, it is necessary to calculate the performance measures. Then the several important measures presented in the above discussion may be computed numerically. Since the equilibrium state probabilities are obtained in simple multiplication form, computations for the performance measures based on the probabilities are significantly simplified.

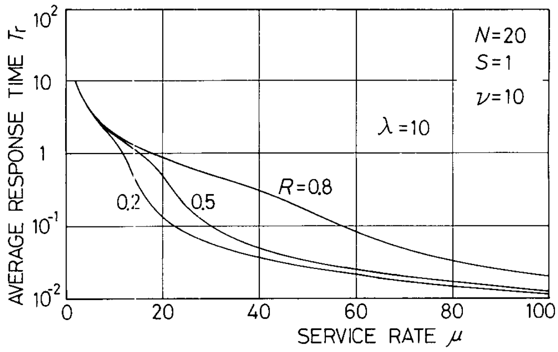

Fig. 4 Average response time ($\lambda=10$).

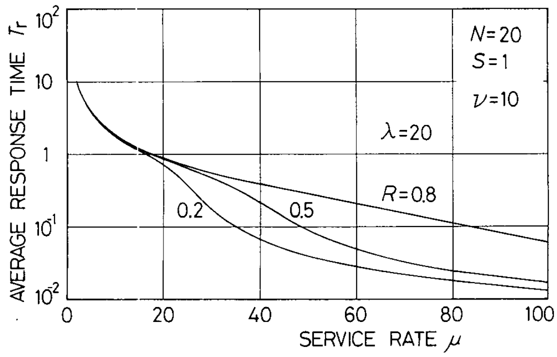

Fig. 5 Average response time ($\lambda=20$).

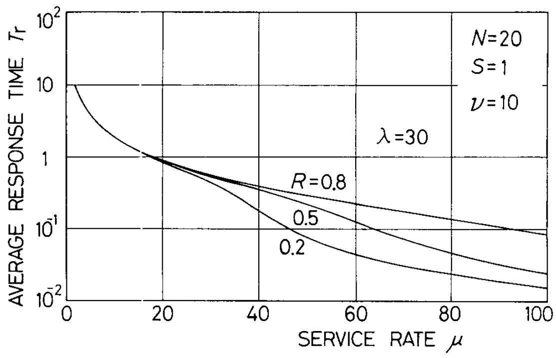

Fig. 6 Average response time ($\lambda=30$).

Figs. 4 to 6 show curves of the average response times which are regarded as the most important performance measures for the system evaluation. All the transverse axes in the three figures indicate service rate of a computer

μ, and it is similar for all following other figures. To make the evaluation clear, the arrival rate of user λ is selected as a parameter, and different values of the rates 10, 20 and 30 are taken as examples for the figures. Although various values can be selected for S, N, λ, ν and μ, the values for the parameters are selected to show only a general tendency of the system performances. The figures show dependencies of the response times on differences of the arrival rates. The curves for the response time are commonly going down with increasing of service rates, and the property is naturally understeed.

In the figures, the remain probability R is selected as a parameter, and various values 0.2, 0.5 and 0.8 are given to the probability. The remarkable properties that rapid decreasing poists appear on each curve of 0.2, 0.5 and 0.8 are shown in each figures. Although more detail considerations are necessary to make the tendency clear, it is basically resulted from introducing of the probability R.

The probability R expresses the situation that many tasks including a first task, a second task and so on are sent by a user and sent to computers sequentially. In such the situation many tasks are repeatedly sent to the computers, and a task sent from a user must wait the summed queue constituted by kartial queue of first sent tasks and partial queue of second or more late sent tasks from other users. The task in the system where one task only is sent from a user, however, is necessary to wait only the queue constituted by first sent tasks. Hence the queue length becomes to be large for large value of R, and appearing point of rapid decreasing is shifted to larger area on service rate axis.

Fig. 7 shows properties of the response time where the number of servers S is taken as a parameter. All the curves are also going down with increasing of service rate, and also have

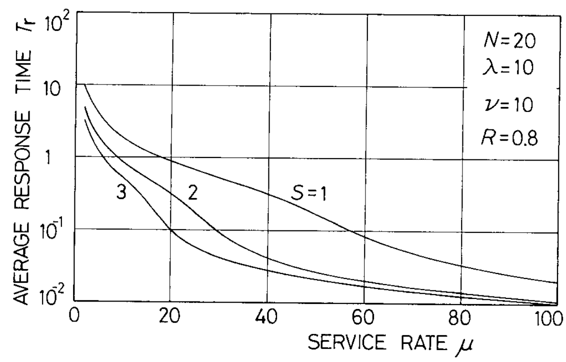

Fig. 7 Average response time ($S=1, 2, 3$).

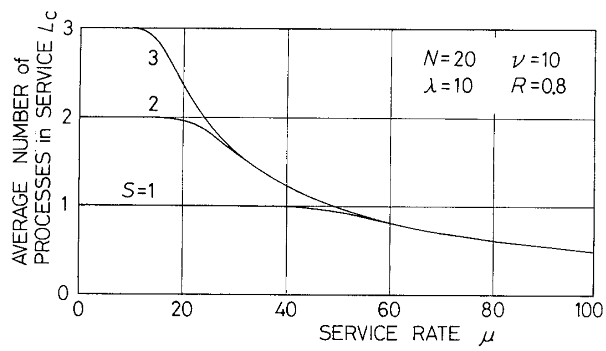

Fig. 8 The average number of tasks in service.

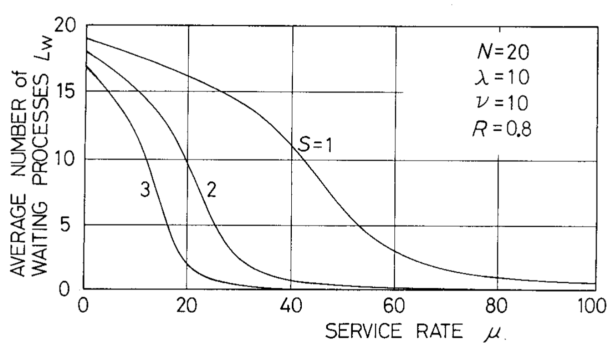

Fig. 9 The average number of aiting tasks.

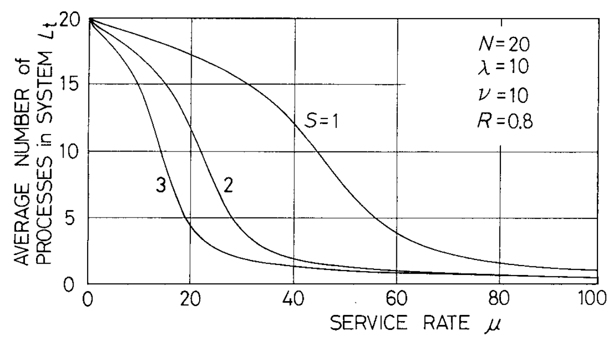

Fig. 10 The average number of tasks in system.

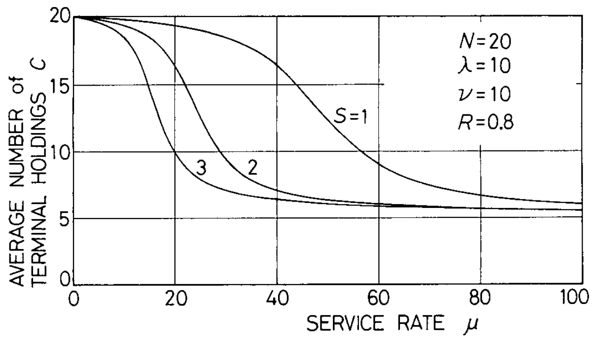

Fig. 11 The average number of terminal holdings.

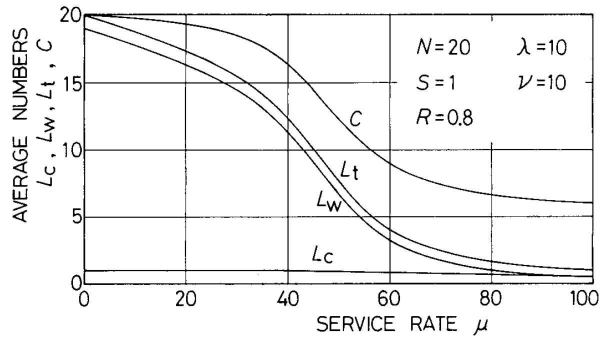

Fig. 12 The average number of tasks in service L_c, waiting tasks L_w, tasks in system L_t and terminal holdings C.

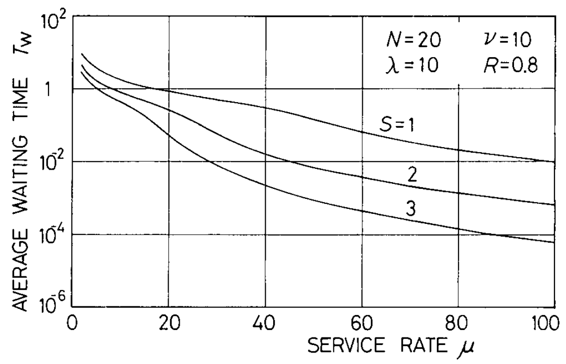

Fig. 13 Average waiting time ($S=1, 2, 3$).

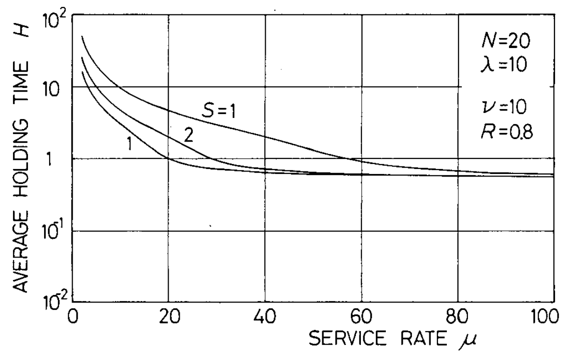

Fig. 14 Average holding time of terminal.

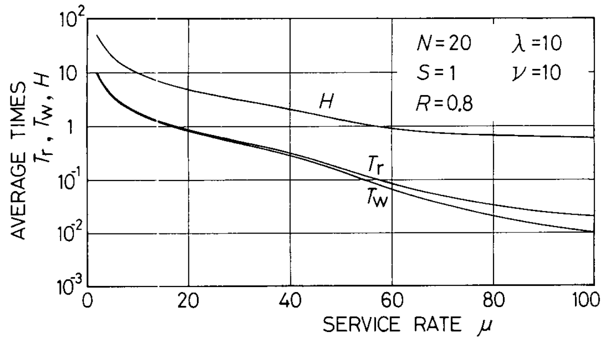

Fig. 15 Average times of response T_r, waiting T_w and holding H.

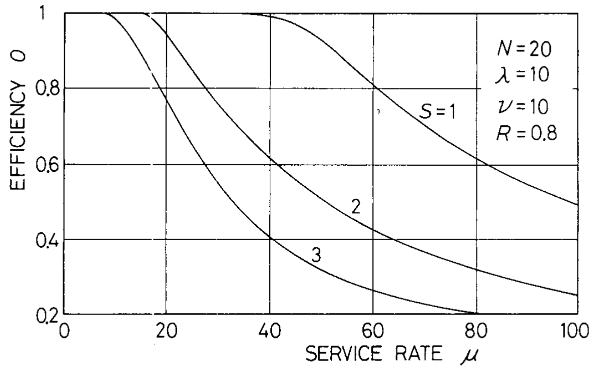

Fig. 16 Efficiency of computer operation.

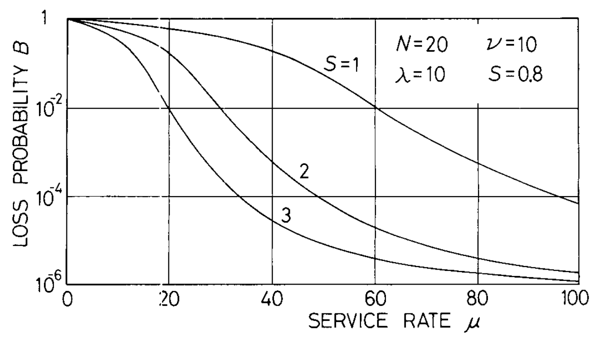

Fig. 17 Loss probability.

rapid decreasing points on them.

Fdgs. 8 to 10 show the average numbers of tasks in service L_c, waiting tasks L_w and total tasks L_t. Fig. 11 shows the average unmber of terminal holdings (the number of users sitting in front of terminals) C. Fig. 12 summarizes the curves of L_c, L_w, L_t and C to compare the properties each other.

Fig. 13 and 14 show the average waiting time

T_W and the average holding time of terminal H. Fig. 15 summarizes the curves of T_W, T_r and H, and shows that the holding time H tapes significantly large value than the response time T_r and the waiting time T_W.

Fig. 16 shows the operating efficiency of computers O and Fig. 17 finally shows the loss probability B.

5. CONCLUSION

A repeat task sending model is presented to analyze the performances of a conversational mode multiple computer system influenced by stochastic working property of users on terminals. A repeat task sending property of a user is characterized by a remain probability R introduced as the probability that the user remains at a terminal after he received his served task. To analyze the performances of system the equations to probabilities for queue sizes are derived, and the solution to the equations is obtained. Expressions of the probabilities have simply multiplication form, and then are useful to simplify numerical computations. Various performance measures, for examples average response time, average holding time of terminal, etc. are given for system design. Numerical results are finally given in many figures to show typical properties of the several performance measures.

REFERENCES

[1] E. Gelenbe "Introduction" *IEEE Trans. Software Eng.*, vol. 17, no. 10, pp. 985-986, 1991.

[2] W. W. Chu, C. M. Sit and K. K. Leung "Task Response Time For Real-Time Distributed Systems With Resource Comtentions" *IEEE Trans. Software Eng.*, vol. 17, no. 10, pp. 1076-1092, 1991.

[3] K. K. Parhi and D. G. Messerschmitt "Static Rate-Optimal Scheduling of Iterative Data-Flow Programs via Optimum Unfolding" *IEEE Tans. Computers*, vol. 40, no. 2, pp. 178-195, 1991.

[4] M. Nakamura, I. Sasase and S. Mori "Two Parallel Queues With Dynamic Routing Under a Threshold-Type Scheduling" *Trasn. IEICE*, vol. E 73, no. 3, pp. 379-385, 1990.

[5] H. C. Lin and C. S. Raghavendra "A Dynamic Load-Balancing Policy With a Central Job Dispatcher (LBC)" *IEEE Trans. Software Eng.*, vol. 18, no. 2, pp. 148-158, 1992.

[6] C. G. Rommel "The Probability of Load Balancing Success in a Homogeneous Network" *IEEE Trans. Software Eng.*, vol. 17, no. 9, pp. 922-933, 1991.

[7] N. S. Bowen, C. N. Nikolaou and A. Ghafoor "On the Assignment Problem of Arbitrary Process Systems to Heterogeneous Distributed Computer Systems" *IEEE Trans. Computers*, vol. 41, no. 3, pp. 258-273, 1992.

[8] S. Q. Li and J. W. Mark "Traffic Characterization for Integrated Services Networks" *IEEE Trans. Commun.*, vol. 38, no. 8, pp. 1231-1243, 1990.

[9] B. A. Blake and K. Schwan "Experimental Evaluation of a Real-Time Scheduler for a Multiprocessor System" *IEEE Trans. Software Eng.*, vol. 17, no. 1, pp. 34-44, 1991.

[10] R. Gusella "A Measurement Study of Diskless Workstation Traffic on an Ethernet" *IEEE Trans. Commun.*, vol. 38, no. 9, pp. 1557-1568, 1990.

[11] R. L. Bagrodia and C. C. Shen "MIDAS: Integrated Design and Simulation of Distributed Systems" *IEEE Trans. Software Eng.*, vol. 17, no. 10, pp. 1032-1058, 1991.

[12] F. Baskett, K. M. Chandy, R. R. Munts and F. G. Palacious "Open, closed and Mixed Networks of Queues with Different Classes of Customers" *J. ACM*, vol. 22, no. 2, pp. 249-260, 1975.

[13] R. B. Cooper "Introduction to Queueing Theory" *The Macmillan Company, New York*, 1972, Eq. (10.3)

[14] C. Fujihashi "Performance Analysis of Multiterminal-Multicomputer System" *IEICE Technical Report*, CPSY 91-2, pp. 13-24, 1991.

[15] J. D. C. Little "A Proof for the Queueing Formula: $L=\lambda W$." *Oper. Res.*, vol. 9, no. 3, pp. 383-387, 1961.

Appendix 1 The Poisson process

In the queueing theory the Poisson process is is used as the most fundamental process. The Poisson process corresponds to the Markov random process. The probability that n tasks arrive in a time interval t in the Poisson process with an arrival rate λ is given by

$$\frac{(\lambda t)^n}{n!} e^{-\lambda t}$$

The similar equation also are obtained for the sending and service processes with ν and μ.