Simulation of Real-Time Processing System—MARS 101

SETSUO OSUGA*

1. *Introduction*

This paper describes the method and results of a simulation of a seat reservation system named MARS 101 in Japanese National Railways.

This system, operating since October, 1963, was designed to receive a lot of calls sent from terminals that were placed in more than one hundred railway stations covering whole country, process them instantaneously and send back immediately to remote terminals.

The simulation was performed while this system was being developed and its results were effectively applied.

MARS 101 system is a dual multi-process computer system. A multi-process computer system is composed of single supervisory computer (SC) and two satellite computers (TC and FC). Each computer and transmission control unit (TRC), that controls communication lines, assume different fractions of respective process of a call. In order to assure the effectiveness of the system with such components, the system designer must consider enough about functional balance among the system units as well as their responce times.

The principal object of the simulation was to establish the optimum condition by sequential tests varying system parameters with the system particularities.

2. *Description of MARS 101*

Fig. 1 shows a block diagram of the system with main flow of informations in it.

There, SC is a central part of the system with the high speed core memory, control unit, arithmetic unit and date channels. The main roles of this are to control the traffic flow of informations, to supervise other processing units through interruption control, and perform such routine works as bill accounting, code conversion, basic file maintenance, error detection and its indication, etc..

TRC controls 96 communication lines concurrently at the maximum, and each datum through respective communication line is directly written into or read out of high speed core memory.

TC is a drum-based processor and contains proper informations to each train such as class, names of stops, termini, etc..

FC is also a drum-based processor and contains all informations of every seat of a train for several days. Provided the train is of long range with many stops, one

This paper first appeared in Japanese in Joho Shori (the Journal of the Information Processing Society of Japan), Vol. 6, No. 1 (1965), pp. 1-11 and Vol. 6, No. 2 (1965) pp. 74-80.

* Institute of Space and Aeronautical Seience, University of Tokyo.

seat of this train has several informations in order that this one seat may be serially used by several short range customers in one operation.

These TRC, TC, and FC as well as magnetic tape controllers (MTC) that are used to back up the system operate independently one another and also independently of SC after receiving the operation command from SC.

In Fig. 1, AS means another multi-process computer system of this dual system. Each computer system processes the same call independently and each processed information is collated between them.

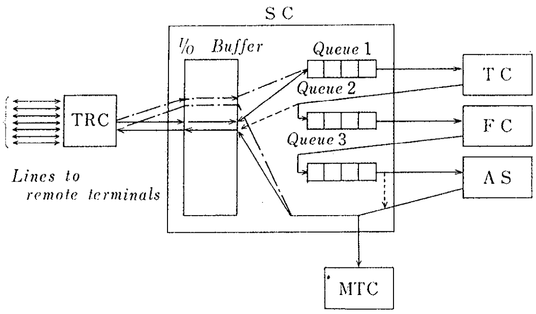


Fig. 1.

3. *Modeling of the System*

Modeling for the simulation was first performed with respect to separate units composing the system, and then a model of the total system was constructed. In consequence, a logical structure of the system model was constituted of eight independent items with respective queues that were connected sequentially. These items generated new events with respectively predetermined intervals after the last events. This inter-event time of each item was definite for some items and indefinite for other items requiring random variable, and whenever new event occured, next inter-event time was prepared by the simulation control program according to inherent algorithm of each item.

4. *Time Control*

Time is a prominent variable of the simulation. There are two methods of time control to be used. In one method, a unit period of time is established and simulation processes all the events that occur in each unit of time and then advances to the next. In another method, on the other hand, the clock is advanced from the time the latest event has occured to the time the next earliest event will occur. In our simulation, the latter method was used from the viewpoint of simulation time. There, to find the earliest event, a time list was used, that contained the residual time to the next event for all items and was looked up to find an item of the smallest residual time amoung them. As any event occured, the content of this list was up-dated.

5. *Generation of Random Number*

Some items in the model contained uncertainties in their inter-event times. Therefore, there needed a way of introducing random variables into the model. In our case, some items needed negative, exponentially distributed random variables and the others required specifically distributed random variables that should be decided from the physical characteristics of the system. The basis for the generation of these random variables is the generation of a series of numbers uniformly distributed in the range (0,1). These numbers were used to generate specifically distributed random numbers.

As stated above, our simulation intended to establish the optimum condition and this was performed through sequential test varying not only several parameters but also the distributions of the random variables assuming variations of characteristics of the devices.

6. *Recorded Data*

Selection of data to be recorded is important. In our simulation, two kinds of data were recorded, current data and statistical data. Current data were recorded in order to visualize the pattern of transition of the system status, and all status informations were printed out whenever a new event occured. In current data, there were included such records as number of repetitions, interevent time, item number that generated the event, queue length of respective item, accumulated time from start point, etc.. By these records, we could well understand how transition pattern of states was influenced by external and internal environments of the system, and this observation was quite useful for development of software system.

Statistical data, on the other hand, included several accumulated data that were later used to investigate system performance. They included following data;

1) Accumulated continuation time of respective queue length for each item (this is shown in this paper as $T_{i\nu}$ where i and ν mean item number and queue length respectively). That could be used to get queue length probailities and total time of the simulation.

2) Effective fraction of above time each item was not in idle state. That was used to get each item's effective and loss time percentages ($T_{i\nu 1}$).

3) Total number of processed calls during the simulation (N).

4) Sum of the times requiring to process respective calls in the system (T').

5) Sum of squares of the times required to process respective calls in the system (T'').

These data were later used and several results were obtained.

7. *Parameters*

During the simulation, several parameters were varied and sequential tests were executed in order to find general trend of the system variation and to establish an optimum system. These parameters were mean input inter-arrival time, TRC's capacity, input and output transmittion time, permissible length of queues in the system, program execution time in SC, and distributions of processing time of TC and FC.

8. *Simulation Program*

Simulation program controls the state of the system and causes the state to change whenever any event occurs and then makes provision for next occurence. Also, the program controls the occurence of the next event according to the current state of the system. In our simulation, the main roles of this program were as follows;

1) To find out the next earliest event,

2) To advance the clock,

3) To generate the random numbers and convert them to specifically distributed random variables,

4) To update the state of the individual item and various control conditions,

5) To update the state of the system common to all items and various control conditions,

6) To monitor the number of repetitions, update the several data tables, print out the data, etc..

9. *Runs*

In order to get useful results and to make effective use of them for understanding the system characteristics, the best combinations of parameters must be chosen. Thus, based on a rough system analysis and several pre-experiments on its results, 150 combinations of parameters were chosen. Number of repetitions of each run was defined by number of generations of random numbers and decided to 5,000. Morever, in order to get steady state results, some repetition cycles of transient state had to be accumulated. There arose a question. How many cycles were necessary to assure to approach the steady state? This seemed to depend on external and internal environment of the system, so we chose several combinations of parameters that were thought to be slow to approach the steady state and executed trial runs. According to their results, adding some safety factor, we decided the maximum number of repetition cycles to 6,000. For simulation, HITAC 3030 that was to be adopted as supervisory computer of MARS 101 was used, and simulation time was about 4 minutes when only statistical data were recorded and about 30 minutes when both current and statistical data were recorded.

10. *The Results*

Using statistical data, various results could be obtained. We were mainly interested in 1) mean process time, 2) process time of individual call and 3) system efficiency.

Mean process time (γ) was obtained from the total number of processed calls (N) and the total time spent in processing them (T). The total number of calls was recorded as one of statistical data and elapsed time was obtained by summing up the queue length continuation times with respect to queue length of each item, that is, $T=\sum_{\nu} T_{i\nu}$. Then $\gamma=T/N$.

Fig. 2 shows some of the relations between γ and number of communication lines controlled by TRC. The figure also indicates transition of the bottle neck point in the system performance from TRC capacity to capacities of other units as TRC capacity increases.

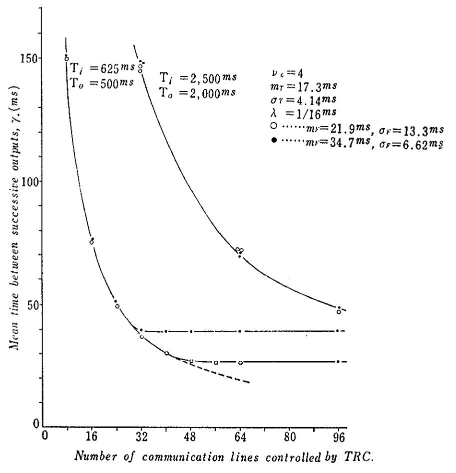
Process time of individual call (γ') was obtained from N and T' that was recorded as one of statistical data, that is, $\gamma' = T'/N$. Standard deviation of process times (σ) was also obtained from N and T'' (Refer to section 6).

Fig. 3 shows the relation between γ (and σ) and TRC controllability again. There, upper flat region shows that saturation occured in the system. As the number of calls that could exist in the system at a time was limited, too many calls could not enter, but overflew. Then inner system state became stable. This was unrealistic, however, because this was the case the input rate was unreasonably high.

Distribution of the calls in the system, that is, queue length probabilities ($p_{i\nu}$ with same suffix as $T_{i\nu}$) was obtained from $T_{i\nu}$ and T as $p_{i\nu} = T_{i\nu}/T$. Fig. 4 is an example of the relations between queue length probabilities and the input rate to the FC queue. From these results, mean queue length and deviation of queue lengths were obtained. On the other hand, efficiency of the system or efficiencies of several components were obtained as ratios of operating times to the total time. Fig. 5 shows an example of the relations of the efficiencies of SC, TC and FC with respect to the TRC capacity.

11. *Conclusion*

There will be no doubt that simulation is a quite powerful method for system design. We have obtained many usefull data by simulation, and by these results



Definitions of symboles are as follows;

T_i, T_o; Input and output transmission time respectively (ms),

ν_o; Permissible queue length,

m_T, σ_T; Mean and standard deviation respectively of process time required in TC (ms);

m_F, σ_F; Mean and standard deviation respectively of process time required in FC (ms);

λ; Input rate (calls/ms)

n; Number of communication lines controlled by TRC,

Fig. 2.

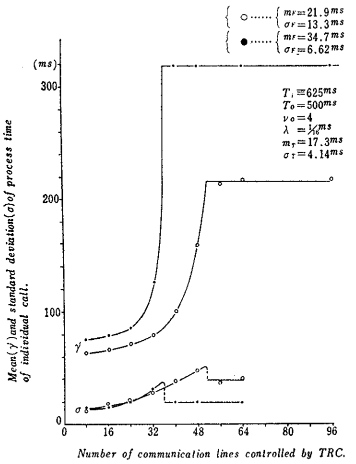


Fig. 3.

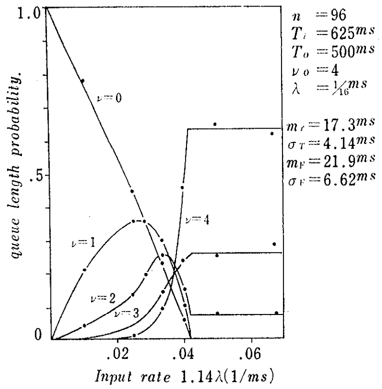


Fig. 4.

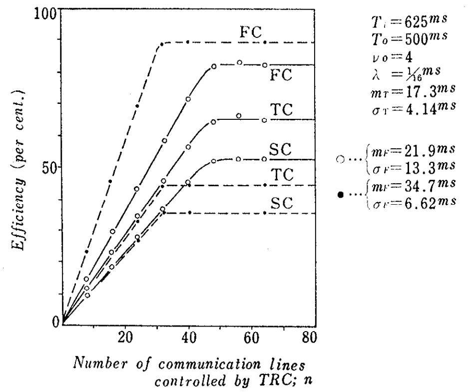


Fig. 5.

we could understand the general trend of the intended system and also got several indications necessary for developing the system. Moreover, these data were thought to be usefull for future evolution of the system.

However, a simulation always requires a lot of computer time. In our case, the data of 150 runs were used but plenty of data of many other runs were given up because these data were judged meaningless or insufficient to be used. Moreover, it takes long time for preparation of a simulation. So, a simulation must be used when we can not expect to obtain good results by any other method, and also, a simulation must be simple enough as long as it fulfills the requirements.

Our purpose of this simulation was to establish the optimum condition and we mainly interested in the mean process time and the system's balance as well as responce time. Though principal object of a simulation may differ as the case may be, the most pertinent model representing faithfully the system characteristics should be established in every case.

Acknowledgment

The author specifically thanks to professor M. Hosaka in University of Tokyo for his many usefull advices for this work, and to J. Abe and M. Asai in Hitachi Co. Ltd. for their assistances.

References

[1] Hosaka, M., Y. Ono and T. Tani, Basic Idea for Actual Time Processing for MARS 101 Unit Seat Reservation System. *The Hitachi Hyoron, 46*, 6 (1964).

[2] Inoue, H., S. Osuga, J. Abe and others, Real Time Programming for MARS 101 Seat Reservation System. *The Hitachi Hyoron, 46*, 6 (1964).

[3] Saisu, K., Automatic Seat Reservation System in J.N.R. *Joho Shori* (*the Journal of the Information Processing Society of Japan*), *5*, 1 (1964).