Interactive Graphic Simulation System

Yukio Kaneda* and Toshio Shimada*

Abstract

This paper describes an interactive graphic simulation system called EGSS by using flowchart input.

This system was designed so that simulation models are built and tested incrementally. The objective of incremental simulation is to improve the efficiency of building models by removing the distinction between the building and testing phases.

INTRODUCTION

The development of simulation languages has made available a variety of powerful capabilities for model building and experimentation.

Our simulator's goal is to remove the distinction between the building and testing phase of system simulation by allowing the user to interact with his model by a graphic device.

When studying a complex problem, it is often helpful to understand its global scheme and then to subdivide it into several parts. Our simulator makes it possible to build a complex system step by step.

THE SIMULATION LANGUAGE

We chose GPSS as a basic simulation language, because it is one of the most popular simulation language and makes possible to

This paper first appeared in Japanese in Joho－Shori (Journal of the Information Processing Society of Japan), Vol. 13, No. 10 (1972), pp. 665～672.

* Computer Systems Section, Computer Division,
Electrotechnical Laboratory, Japanese Government

build a program by a block diagram.

STRUCTURE OF THE SIMULATOR

Our simulator has three phases, which are

modeling phase : to build a simulation model by block diagram,

simulation phase : to execute the simulation program, and

output phase : to display simulation results on a graphic device.

A user can switch these three phases freely to carryout system simulation.

CONCEPT OF BOX BLOCK

In the real simulation model, its object program often consists of more than several hundreds of GPSS blocks, though our graphic device can only display less than 20 blocks in one frame.

To avoide this problem, we introduced a new type of block called BOX block and realized a hierarchically structured model building approach.

One can divide the simulation model into several parts.

Each part may be further subdivided until each part can be displayed in one frame of the graphic device. Every part (every frame) is associated with different BOX block. (Fig. 1)

In our system, a simulation model is structured into a tree where non-terminal nodes are BOX blocks and terminal nodes are ordinary GPSS blocks.(Fig. 1)

Each BOX block represents all blocks which belong to its subtrees, the root BOX represents the entire simulation model.

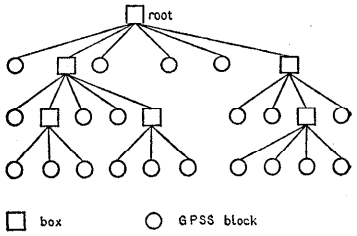

Fig. 1 Tree structure of a model.

MODELING PHASE

At the starting of modeling phase, our simulator displays the BASE frame, which corresponds to the root BOX block.(Fig.2)

BASE

DELETE	EXECUTE	CREATE	PAGE
ALTER	PARAM	SWAPOUT	CONNECT
END			

Fig. 2 Base frame

A user can display any GPSS block in any position of the frame by light pen positioning and can input parameters for that block from the console typewriter.

Before the display frame becomes full of blocks, an operator (model builder) must create and name new BOX block.

After that, he can continue the model building process using new frames for the new BOX block. (Fig. 3)

Modification of the simulation model can be easily done at any time in model building phase using the light pen and console typewriter. (Fig. 3)

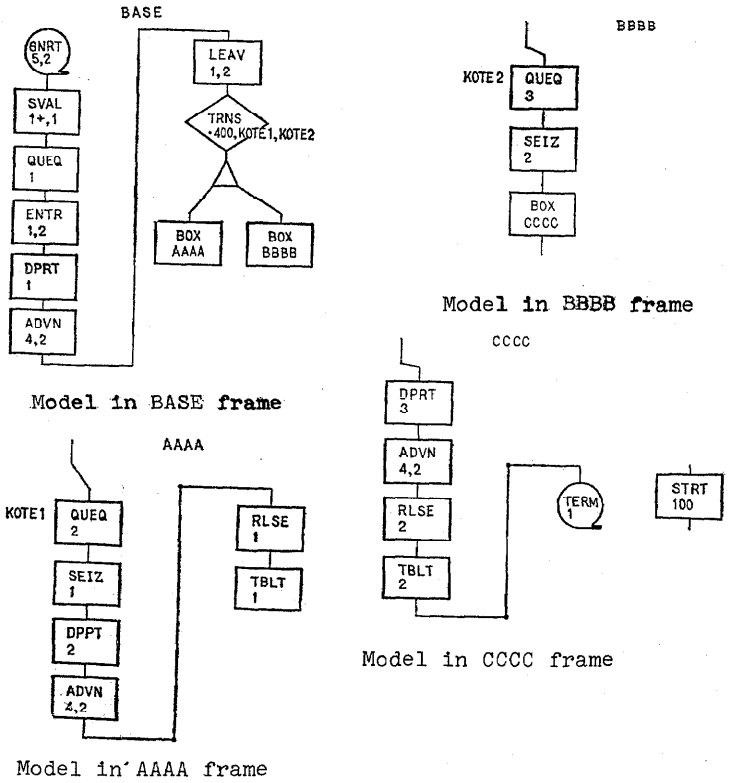

Fig. 3 Example of a Model.

SIMULATION PHASE

When a user commands the system to execute the simulation program, the simulator takes the following two steps.

1 Generation of the program by tracing the model tree (Fig. 4)

2 Execution of the generated program.

After these two steps, OUTPUT program is activated and the result of the simulation is displayed on the graphic device. (Fig. 5, Fig. 6)

```
1       TABLE       M1, 4, 1, 15
2       TABLE       M1, 4, 1, 14
        GENERATE    5, 2
        SAVEVALUE   1+, 1
        QUEUE       1
        ENTER       1, 2
        DEPART      1
        ADVANCE     4, 2
        LEAVE       1, 2
        TRANSFER    ,400, KOTE1, KOTE2
KOTE1   QUEUE       2
        SEIZE       1
        DEPART      2
        ADVANCE     4, 2
        RELASE      1
        TABULATE    1
        TERMINATE   1
KOTE2   QUEUE       3
        SEIZE       2
        DEPART      3
        ADVANCE     4, 2
        RELEASE     2
        TABULATE    2
        TERMINATE   1
        START       100
```

Fig. 4 Generated program.

PARTIAL EXECUTION

A user can command the simulator to generate a subprogram which is contained in a BOX block and execute it.

By associating each program part with each BOX, one can test any portions of his model.

After building and testing each part, one can combine them into the complete model.

```
BLOCK                    ****  GPSS     OUTPUT     ****
NUMBER    *LOC    OP          A, B, C, D, E, F, G          COMMENTS     CARD#
                  JOB                                                   00001
                  REALLOCATE 200 K                                      00002
          *                                                             00003
          *                                                             00004
          *                                                             00005
          *                                                             00006
          *                                                             00007
          1       STORAGE     4                                         00008
          2       TABLE       M 1, 4, 1, 15                             00009
          1       TABLE       M 1, 4, 1, 15                             00010
00001             GENERATE    5, 2                                      00011
00002             SAVEVALUE   1+, 1                                     00012
00003             QUEUE       1                                         00013
00004             ENTER       1, 2                                      00014
00005             DEPART      1                                         00015
00006             ADVANCE     4, 2                                      00016
00007             LEAVE       1, 2                                      00017
00008             TRANSFER    , 400, KOTE 1, KOTE 2                     00018

 INPUTLIST                                                  TOP   NEW   STOP
```

Fig. 5 Example of output.

**** GPSS OUTPUT ****

QUEUE NUMBER	MAXIMUM CONTENTS	AVERAGE CONTENTS	TOTAL ENTRIES	ZERO ENTRIES	PERCENT ZEROS
1	1	0.000	101	101	100.00
2	1	0.050	60	50	83.33
3	1	0.029	40	32	80.00

QUEUE NUMBER	AVERAGE TIME/TRANS	AVERAGE TIME/TRANS	TABLE NUMBER	CURRENT CONTENTS
1	0.000	0.000	0	0
2	0.433	2.600	0	0
3	0.375	1.875	0	0

QUEUE

**** GPSS OUTPUT ****

FACILITY NUMBER	AVERAGE UTILIZATION	NUMBER ENTRIES	AVERAGE TIME/TRANS	SEIZING TRANS. NO.	PREEMPTING TRANS. NO.
1	0.495	60	4.283	0	0
2	0.305	40	3.950	0	0

FACILITY TOP NEW STOP

Fig. 6 Example of output.