NCJRS

This microfiche was produced from documents received for inclusion in the HCIRS data base. Since NCIRS cannot exercise control over the physical condition of the documents submitted, the individual frame quality will vary. The resolution chart on this frame may be used to evaluate the document quality.



Microfilming procedures used to create this fiche comply with the standards set forth in 41CFR 101-11.504

Points of view or opinions stated in this document are those of the author(s) and do not represent the official position or policies of the U.S. Department of Justice.

U.S. DEPARTMENT OF JUSTICE LAW ENFORCEMENT ASSISTANCE ADMINISTRATION NATIONAL CRIMINAL JUSTICE REFERENCE SERVICE WASHINGTON, D.C. 20531

Date filmed

12/23/75

A SIMULATION MODEL OF POLICE PATROL OPERATIONS: PROGRAM DESCRIPTION

OF HOUSING AND URBAN DEVELOPMENT AND FOR THE CITY OF NEW YORK

PETER KOLESAR WARREN E. WALKER



PREPARED FOR THE DEPARTMENT

R-1625/2-HUD/NYC FEBRUARY 1975

This report was prepared as an account of work sponsored by the United States government, Office of Policy Development and Research, Department of Housing and Urban Development (HUD), under Contract H-2164, and by the Police Department of the City of New York. Neither the United States nor HUD, nor any person acting on behalf of HUD, nor the City of New York, makes any warranty, expressed or implied, or assumes responsibility for the accuracy or completeness of the information herein, or assumes liability with respect to the use of any information or method herein.

A SIMULATION MODEL OF POLICE PATROL OPERATIONS: PROGRAM DESCRIPTION

PREPARED FOR THE OFFICE OF POLICY DEVELOPMENT AND RESEARCH, DEPARTMENT OF HOUSING AND URBAN DEVELOPMENT,

AND FOR THE CITY OF NEW YORK

The New York City-Rand Institute is a nonprofit research institu-tion founded, as its Articles of Incorporation state, "... primarily to conduct programs of scientific research and study, and provide reports and recommendations, relevant to the operations, planning or admin-istration of the City of New York." The Institute was established in 1969 by the City of New York and The Rand Corporation as a center for the continuing application of scientific and analytic tech-niques to problems of urban life and local government. It is governed by a Board of Trustees appointed jointly by the City and Rand.

Published by The Rand Corporation

PETER KOLESAR WARREN E. WALKER

R-1625/2-HUD/NYC **FEBRUARY 1975**

3**6**2

THE

RAND

INSTITUTE

NEW YORK CITY



545 MADISON AVENUE NEW YORK NEW YORK 10022 (212) 758-2244

This Report is designed to provide computer programmers and systems analysts with detailed documentation of a computer program developed at The New York City-Rand Institute for simulating the activities of police patrol cars in a city or a region of a city. The development of the simulation program was supported under a contract with the New York City Police Department. Its generalization and documentation was supported by a contract with the Office of Policy Development and Research of the Department of Housing and Urban Development.

The HUD contract has as its objective the development, testing, and documentation of methods to improve the allocation of resources in municipal emergency service agencies. Making these techniques widely available should ultimately result in significant improvements in the delivery of municipal emergency services.

It is presumed that the reader of this Report knows something about police patrol operations and terminology, and quite a bit about computer simulation. A companion report is being prepared that will describe the simulation and its uses for police department administrators and other city officials:

R-1625/1-HUD, A Simulation Model of Police Patrol Operations: Executive Summary, Peter Kolesar and Warren E. Walker, The New York City-Rand Institute, forthcoming.

These two reports are part of a series that documents several different deployment models for police, fire, and ambulance services, and describes the application of the models in several cities. Further information can be obtained from The New York City-Rand Institute.

-111-

PREFACE

The computer simulation model described in this report was written to study the police patrol operations of the New York City Police Department. The structure of the program, however, is sufficiently general that, with only minor modifications, it could be used to study the operations of any large metropolitan police department.

The simulation is designed to examine the effect of changes in a police department's deployment of its patrol resources. Among the policy options that can be tested are: changing the number of patrol units on duty; changing the boundaries of the regions assigned to particular patrol units for patrol and for dispatch to calls; and changing dispatching procedures. Alternative policies may be compared based on a wide range of performance measures provided by the simulation. These include the delay between when a call for service is received and when it is dispatched, the delay between when a call is dispatched and when a patrol car arrives at the scene, and the workload of individual patrol units.

The program is written in SIMSCRIPT II.5. This report describes each of the routines in the program and the program's data requirements in sufficient detail that a user who is familiar with SIMSCRIPT should be able to run or modify the program. Flow charts, a complete program listing, sample data decks, and program outputs are all included.

-v-

The New York City-Rand Institute's Fire Operations Simulation, designed by Grace Carter and Edward Ignall, served as a model whose success we hoped to imitate. Although the details, problems, and programming language of the Police Patrol Simulation are different, many of the design concepts of Carter and Ignall served us in good stead. Captain Daniel Cawley of the New York City Police Department checked the reasonableness of many of the assumptions that we built into the model. Jack Hagouel and Harry Elam made modifications to our original design, which were incorporated in the version reported here. James Tien's detailed knowledge of dispatching operations was also helpful. The final report was considerably improved by the constructive, comments of Grace Carter and John Schank of the Rand Corporation, and Calvin Clawson and Samson Chang of the Seattle Police Department, who reviewed an earlier draft.

Finally, we acknowledge the graciousness, patience, and dedication of the officers and patrolmen of the New York City Police Department in whose patrol cars we rode while learning something about what occurs in the real world of the streets of New York City.

ACKNOWLEDGMENTS

| PREFACE | |
|----------|----------------------------------|
| SUMMARY | |
| ACKNOWL | EDGMENTS |
| LIST OF | FIGURES |
| LIST OF | TABLES |
| a | |
| Section | |
| I. I | NTRODUCTION |
| II. D | ESIGN CONSIDERATIONS |
| | Policy Options |
| | Measures of Performance |
| | Job Stream |
| דדד ייי | |
| TTT T | HE STRUCTURE OF THE SIMULATION I |
| | Patrol Resources |
| | Movement of Patrol Unite |
| | The Job Stream |
| | The Dispatching Activity |
| | ine proparenting neervicy |
| IV. E | WENTS AND SUBROUTINES |
| | MAIN |
| | EVENT FOR JOB.ENTRY |
| | Dispatch Subroutines: DISPR1, 1 |
| | DISPR4, DISPR5 |
| | ROUTINE TO INTERPOLATE.LOCATION |
| | ROUTINE TO CANCEL.SECTOR.RETURN |
| | ROUTINE TO PREEMPT (K) |
| | ROUTINE TO ASSIGN (K,J) |
| | ROUTINE TO SCHEDULE CAR.ARRIVAL |
| | ROUTINE TO CALCULATE.DISTANCE (|
| | EVENT FOR ARRIVAL.AT.SCENE(K) |
| | ROUTINE FOR END.CALL.SCHEDULING |
| | EVENT CALL.END GIVEN K |
| | ROUTINE TO REASSIGN.CAR.TO.JOB |
| | ROUTINE TO PLACE.CAR.ON.PATROL |
| | EVENT RETURN.TO.SECTOR GIVEN K |
| | EVENT OUT OF SERVICE GIVEN K SA |
| | EVENT FOR END. OF. SIMULATION . |
| Vт | ΗΕ ΤΝΡΙΙΤ ΠΑΤΑ |
| ¥. 1 | The Initialization Deck (GO SYS |
| | The Job Stream (SIMUO4) |
| | An Example |

| • | • | • | • | | | •. | • | • | ₽ ", ¹ | ÷. | | • | 1∉2 | • | | • | iii |
|------------|--------------|------------|--------------|------------|------------|--------|------------|------------|--------------------------|------|-----|-----|-----|------|-----|-----|------|
| • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | • | v, |
| • | • | • | • | • | • | • | • | • | • | • | | • | • | • | | • | vii |
| • | | • | • | • | • | , • | • | • | •. | • | • | • | • | °.•* | • | • | xí |
| • | • | • . | | | • | | • | • | • | | • | • | • | • | • | • | xiii |
| | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | |
| • | • | • | • | • | • | • | • | • | •' | • | • | • | • | • | • | • | 1 |
| • | | | | | • | | | | | | | | | | | | 4 |
| • | • | • | • | • | • | • | • | • | • | | | • | | • | • | | 4 |
| ÷ | • | . • | • | • | • | ٠ | ٠ | • | • | • | • | • | • | • | • | • | 5 |
| 4 | ٠ | ٠ | • | • | • | • | • | • | • | ٠ | • | • | • | • | • | • | 5 |
| N | M | DDI | EL | A | ND | ΤI | ΗE | J | OB | S | rr | EAI | 1 | • | • | ÷ | 7 |
| • | ٠ | ٠ | | • | • | • | • | 4 | • | • | • | ٠ | . • | | • | • | ्र 7 |
| • | •, | ٠ | • | ٠ | • | • | • | • | • | ٠ | • | ٠ | • | . • | ٠ | • | 0 |
| • | • | : | · • . | | | | : | | • | • | 1 | • | | • | | • | 12 |
| • | • | • | | | • | • | • | • | . 10 | | | • | | | | • | 14 |
| | | | | | | | | | | | | | | | | | |
| • | .• | • , | ٠ | • | ٠ | • | •, | • | • | • | • | ۲ | ٠ | • | ٠ | ٠ | 16 |
| : | • | •, | • | | • | • | • | • | | • | • | • | • | • | • | • | 23 |
| . 9 | D | ESI | PR2 | 2, | D | LSI | R. | 3, | | • | | · | • | • | •, | • | |
| • | • | • | | ٠ | • : | • | ٠ | • | • | • | • | | • | • | ٠ | • | 25 |
| 'I(| DN. | G] | | EN | K | Aì | ND. | J | • | • | • | . • | •. | • | ٠ | ٠ | 31 |
| UI | KIN . | G. | | ΞŊ | ĸ | ٠ | • | • | .• | ٠ | ٠ | • | • | ٠ | .• | • | 32 |
| • | ٠ | | • | ٠ | ٠ | ٠ | • | • | • | • | ٠ | • | • | • | • | .• | 22 |
| $\dot{v}l$ | | (v | т, | ` | ٠ | 4 | • | • | .• | • | • | • | • | • | • | • : | 33 |
| • • न | сл | | , U , 7N | ' ¥' | • | v. | • | · v | , · / | | • • | v ? | • | • | • | \$ | 22 |
| ñ | 0. | | 771 | 42. | - 9 | | و | | | 1111 | | 1 4 | | • | • | • | 3/ |
| י. דו | JC. | ċ | • | • 7N | v | | т. | • | TD. | • | • | • | • | • . | • | • | 36 |
| 11 | 10 | 63 | . • 1 | 111 | , K | m | LU | DU |) IC | ٠ | • | • | Ŷ | • . | ٠ | • | 36 |
| • | , * , | דדי | • 1771 | лт | , , | • • | . . | т * | • | • | | • | • | • | • | • | 30 |
| | | ידר דדי | / E4 7178 | 1 N T T | \ £ | TIAT | , , | | • | • | ٠ | • | • | •, | • | .• | 20 |
| 01 | | μT/ | (Er | 1 1 | 5 | ٠ | ٠ | ٠ | • | • | • | • . | • | • | • | • | |
| 1 | 5. 9 J T | • | • | • | * | * | * | • | * | • | • | • | ٠ | • | • | ٠ | 40 |
| , ì | A | / Lr | ιĢ | 11 | iĽ | EV | E | NT. | NU | 111 | LUI | Ľ | ٠ | .* | • | ٠ | 40 |
| • | • | • . | • | • | • | • | ٠ | ٠ | • | • | • | • | • | • | • , | • | 41 |
| • | • | | | | • | | •. | | | • | • | ÷ | • | | • | • | 47 |
| SY | S1 | N) | ł | • | | • | • | • | • | | | | • | • | • | • | 47 |
| | | • | | • | | | | • | • | • | | | • | | • | | 50 |
| • | | | | • | • | | • | | | | | • | • | | • | • | 51 |
| | | 1 | | - | | - | | | | · · | | | - | 2 | | | |

ð

| VI. OUTPUT REPORTS | 55 |
|--|------|
| (1) Car Activity Summary | 55 |
| (2) Queueing Statistics | 56 |
| (3) Response Time Histograms | 56 |
| (4) Neighborhood Activity Summary | 57 * |
| (5) Distribution of Car Availability | 57 |
| (6) Number of Precinct Cars Sent to Calls | 57 |
| | |
| Appendix | |
| A. LISTING OF THE SIMULATION PROGRAM | 67 |
| B. SAMPLE CONTROL CARDS AND INITIALIZATION DECK FOR THE 71ST | |
| PRECINCT IN NEW YORK CITY | 81 |
| C. SAMPLE JOB STREAM | 89 |
| D. A DEFINITION OF EACH GLOBAL SIMULATION VARIABLE | 93 |
| | |
| REFERENCES | 98 |

~x-

| 1. | An Example of How Neighborhoods an |
|-----|---|
| 2. | Flow Chart for Priority 1 Incident |
| 3. | JOB.ENTRY |
| 4. | The Structure of the Dispatcher's |
| 5. | ARRIVAL.AT.SCENE |
| 6. | CALL.END |
| 7. | OUT.OF.SERVICE |
| 8. | Map of 71st Precinct - Brooklyn . |
| 9. | Computer-Generated Map of Tax Bloc 71st Precinct |
| 10. | Listing of the Input Data: Block |
| 11. | Listing of the Input Data: Neight |
| 12. | Car Activity Summary |
| 13. | Queueing Statistics for Priority (|
| 14. | Response Time Histograms |
| 15. | Neighborhood Activity Summary |
| 16. | Distribution of Car Availability |
| 17. | Number of Precinct Cars Sent to Ca |

. .

-xi-

FIGURES

| nd | Se | ect | 501 | rs | aı | re | SŢ | pe | cij | Eie | ed | • | • | • | • | • . | 10 |
|------|-----|-----|-----|-----|-----|-----|-----------|-----|-----|-----|-----|-----|-----|----|-----|-----|----|
| t | · | • | .4 | • | • | • | • | • | ı, | • | • | | ٠ | ٠ | ٠ | ٠ | 18 |
| • | • | • | ٠ | • | • | • | • | • | • | • | • | • | | • | • | • | 24 |
| 01 | :de | ere | eđ | Li | lst | : I | <i>TD</i> | JAC | CEN | νT. | C Į | \R | 5() | [) | • | • | 26 |
| • | • | • | • | • | 3 | • | • | • | • | • | | • | • | • | • | • | 35 |
| • | • | . • | • | • | • | • | ٠ | • | • | • | • | • | • | •. | • | • | 37 |
| • | • | • | • | • | | • | • | • | • | • | • | • | | | | • | 42 |
| | | | • | • | • | • | • | • | • | • | ٠ | • | • | ٩ | . • | •• | 52 |
| cks | 3 8 | ano | 1 : | Sed | cto | ors | 5 1 | Ĺ'n | tł | ne | | | | | | | |
| • | • | • | .• | • | • | e | ÷ | + | • | • | • | • | | • | • | • | 53 |
| . Da | ata | а | • | • | • | • | • | ٠ | • | • | | • | ٠ | • | •., | • | 58 |
| ьоз | cho | 200 | i I | Dat | za | • | • | • | . • | • | • | ٠ | ٠ | • | • | ÷ | 59 |
| • | • | • | • | • | • | • | | • | • | • | • | • | • | 0 | • | | 60 |
| C1a | ass | 3 3 | 3 | • | • | • | • | • | | • | • | . • | • | • | | • | 61 |
| • | é | | • | ٠ | • | • | | • | • | • | • | | • | • | • | • . | 62 |
| • | | • | | • | • | | • | • | • | • | • | • | | | • | | 63 |
| • | | • | • | • | • | | • | • | • | • | • | | | • | • | • | 64 |
| all | Ls | • | ٠ | • | • | • | • | • | • | • | • | • | • | • | • | • | 65 |
| | | | | | | | | | | | | | | | | | |

Attributes Associated with the Ent
 Sample Simulation: Car Parameters
 Sample Simulation: Neighborhood E

د سرد مر د بر د مردد بازم قرر

E G

-xiii-

TABLES

| tity | "BLOCK | -11 | ٠ | • | • | • | • | • | • | • | • | • | 8 |
|------|--------|-----|-----|---|---|---|--------|---|---|----|---|---|----|
| tity | "NBD" | • | • | • | • | • | ٠ | • | • | • | • | • | 8 |
| tity | "CAR" | • | 5 | • | • | • | : • | • | • | • | • | • | 11 |
| tity | "JOB" | • | • , | • | • | • | • | • | • | • | • | • | 14 |
| s . | • • • | • | • | ٠ | • | • | • | • | • | .• | • | • | 54 |
| Data | ••• | • | • | • | • | • | • | • | • | • | | • | 54 |

The simulation program described in this Report was written to provide Simulation, in the sense that it is used here, is an imitation through

the New York City Police Department with a tool for evaluating the effects of various changes in the deployment of its patrol resources. Although designed to reflect patrol operations in New York City, the structure of the program is sufficiently general so that it can be used, with minor modifications, to study the operations of any municipal police department. time of the events that occur during police patrol operations. The program that carries out the simulation maintains, inside the computer, a map of the region being simulated, on which it keeps track of the locations of the incidents requiring police service and the locations of the simulated patrol units. A simulation run imitates patrol activities over a fixed time period, say a series of tours of duty on a series of days. The user provides the computer program with specifications of conditions at the start of the time period being simulated, including the number and locations of the patrol units and the operating and dispatching rules being followed. Then, using an internal timing mechanism and a list of "future" jobs, the program carries out the assignment of patrol cars to jobs, their travel to the jobs, the queueing of calls, if any, and stores statistical summaries of response times, patrol availabilities, workloads, etc., that provide analysts with important information with which to judge the effectiveness of the deployment policies being studied. When the simulation is over, the program prints reports containing the statistics summarizing this information. Different deployment strategies or policies can thus be tried out and compared using the simulation as a kind of "pilot plant." Based on the simulation results, analysts and managers can quickly and economically eliminate options that are clearly bad while more attractive policies can be selected for further analysis and eventual implementation.

The purpose of this Report is to provide programmers and analysts with detailed documentation of this computer program, known as The New York City-Rand Institute Police Patrol Simulation Model. It is presumed that the reader knows something about police patrol operations and quite a bit about simulation. A companion document [7] * will provide an overview of the simula-

I. INTRODUCTION

Figures in square brackets identify references located at the end of

-1-

this document.

tion from the managerial point of view. It will describe the purpose of the model, the costs associated with using it, and provide other information that will be helpful in determining when the simulation should be used and when it should not be used. Readers interested in improving their knowledge of simulation itself can consult a general reference such as [2], [3], or [8].

-2-

The programming language used for the simulation is SIMSCRIPT II.5 [6]. With the program listing and other information provided in this Report, anyone who has access to a SIMSCRIPT II.5 compiler and is familiar with its programming conventions should be able to use the program. Machinedependent details of the use of SIMSCRIPT II.5 on the IBM 360 or 370 series of computers can be found in [5]. The simulation program is written as a set of modular subroutines. Each of the subroutines simulates one aspect of the operation of the system (e.g., dispatching, arrival of a patrol car at the scene of an incident, etc.). The user can carry out many applications using the simulation by simply assigning values to some input parameters. But for some applications, one or more of the subroutines may have to be rewritten. For experiments done for the New York City Police Department we have created several versions of some subroutines, but, for simplicity in this Report, only one version of each subroutine is presented in detail.

The simulation has been tested for both internal and external validity. Internal validation is the process of confirming that the computer program is a logically correct translation of the hypothesized model. We performed the internal validity check by comparing results from the simulation with the output from a queueing model that was a good approximation of the system being simulated. The results of this comparison are described in [4]. External validation is the process of confirming that an inference derived from the simulated system is correct for the actual system. For example, if the simulation has external validity and it shows that a certain deployment policy is better than another policy, we can be sure that it will turn out to be better if it is actually implemented. The external validity check of this simulation model was made by, first, gathering detailed data on actual activities of patrol cars in one region of New York City, and, then, running the simulation model using the observed jobs as input. Comparisons were made between performance measures produced by the simulation and the corresponding actual performance measures. Details of this validation are given in [1].

In Section II of this Report we discuss some of the general considerations that led us to the approach we took in designing the simulation. Section III explains the structural elements of the model. In Section IV the logical flow of the program is shown and each event and subroutine is explained in detail. Section V defines each piece of data required as input for the simulation, and Section VI describes the output statistics. The program listing in Appendix A represents one complete version of the simulation. Appendixes B and C contain a sample initialization deck and sample job stream that can be used to test the program and to serve as

a guide for preparing new input data. Appendix D defines each of the global variables used in the simulation.

-3-

II. DESIGN CONSIDERATIONS

-4-

POLICY OPTIONS

The simulation has been designed to provide a police department with the capability for evaluating a wide range of alternative patrol and dispatch policies. Among the policy options that can be tested are:

- (a) changing the number of patrol units on duty;
- (b) changing sector boundaries and sector assignments of patrol units;
- (c) changing dispatching procedures, e.g., using new nomination lists, new priority rules, unit and job location information in unit assignment.

To test a new policy of type (a) or (b) only the initialization data deck has to be changed. Many dispatching changes can be implemented simply by changing input parameters. Some changes, however, may require writing a new subroutine. We decided that this approach was preferable to trying to write one subroutine that would accommodate all possible dispatching rules, since the number of such rules is extremely large. The simulation has been designed to minimize the effort required to code the new routines.

Separate subroutines are used not only for the decision rules, but for every identifiable submodel of the simulation. Although this structure results in many short subroutines, it increases the flexibility of the program and the ease with which it can be modified.

It is wise to make the first simulation run with the model representing the current practices in the department as closely as possible. This will enable the model to be calibrated (e.g., the travel distance function could be modified or the average response velocities adjusted). It will also provide a "base case" for evaluating new deployment policies.

It should be pointed out that, because of the assumptions about travel velocities, travel distances, incident durations, etc., in the simulation, the absolute numbers representing the results of the simulation are not to be regarded or used as facts, although they will probably be good estimates of actual values. The most important use of the simulation is as a tool for comparing alternative policies by noting the differences in their performance. For example, if the simulated average response times under one policy are significantly better than under another policy, then the same comparative advantage will probably carry over to the real world.

MEASURES OF PERFORMANCE

The performance characteristics produced as output by the simulation fall into the following three categories:

~5~

- priority and by sector.
- to calls, are displayed.

These performance measures are only proxies for measures of patrol effectiveness associated with the goals of a police department; smaller response times and queueing delays and higher patrol availabilities should yield better police service. But, since the nature of the relationships between these performance measures and crime suppression, arrests, and other primary goals of the patrol force are not known, we have not attempted to incorporate them in the simulation.

JOB STREAM

The simulation program and the system parameters that specify the geography, number of cars, sectors, etc., can be viewed as a "black box." Into this box is "played" a previously created sequence of incidents or jobs in chronological order. This sequence, called the job stream, resides on disk or magnetic tape and contains, for each job, its entry time, location, duration, and priority. The playing of this job stream into the simulation model is analogous to the playing of a magnetic tape "containing" music into a tape recorder. As the tape recorder physically transforms the magnetic signals into sound, the simulation model mathematically and logically transforms the input jobs into a set of output performance statistics.

The generation of the job stream has been purposely kept separate from the structure of the simulation model itself. There are several good reasons for doing so:

(1) It enables the user to generate job streams from a wide variety

• Response Time Measures: The mean, variance, and distribution of response times (actually, travel times) are gathered by job

• Queueing Delays: The mean, variance, and distribution of the dispatching queue sizes and waiting times are displayed by job priority. • Car Activity: The proportion of time each patrol unit spends on patrol, working on jobs, etc., as well as the mean, variance, and distribution of the number of precinct cars available to respond

of sources. For example, actual job histories, projections of future call patterns, or results of probabilistic models of call generation can all be used to provide job streams to the simulation.

- (2) It conserves computer time, since, when rerunning the same stream under several different deployment options, the jobs do not have to be regenerated for each simulation run.
- (3) It enables some of the statistical analysis of the job stream to be done outside the simulation itself -- and therefore to be done more economically.

III. THE STRUCTURE OF THE SIMULATION MODEL AND THE JOB STREAM

-7-

We now discuss in turn how the geography, patrol resources, and operating and deployment rules are represented in the simulation. We conclude the section with a description of the job stream and the way in which patrol units are dispatched.

GEOGRAPHY

The simulation models a region as a collection of discrete points. called *blocks*, at which calls for service occur. The blocks may correspond to the centroids of city blocks or census blocks, but they need not. The blocks are assigned grid coordinates according to a rectangular coordinate system whose axes should be chosen parallel to the predominant street directions. All incidents (jobs) occur at these points and all patrol units are located with reference to this coordinate system. Each block in the region is assigned an internal reference number between 1 and N.BLOCK (the total number of blocks in the region), and belongs to the class of permanent entities called BLOCK. * Table 1 describes the attributes associated with the permanent entity BLOCK.

The blocks are aggregated into N.NBD nonoverlapping neighborhoods, which must be assigned sequence numbers from 1 to N.NBD (each block belongs to exactly one neighborhood). Each neighborhood belongs to the class of permanent entities called NBD. Table 2 describes the attributes associated with the permanent entity NBD.

PATROL RESOURCES

The program simulates the activities of patrol cars, including both "sector cars" and supervisory or special cars. Each car is represented as a permanent entity (in the class called CAR), and is described by several attributes, including its name, sector responsibilities, current location, and the type of job on which it is working. The patrol area assigned to a particular car is called its sector (also called a beat in some cities). A car's sector is composed of the set of all neighborhoods for which that car has patrol responsibility. Each sector will be assigned one or more

-6-

*See [6] for an explanation of entities and attributes in SIMSCRIPT II.5.

Table 1

-8-

ATTRIBUTES ASSOCIATED WITH THE ENTITY "BLOCK"

| Attribute | Description |
|-----------|--|
| TAXNO | The external identification number associated with the block |
| NBDID | The internal reference number of the neighborhood to which the block belongs |
| XCORD | The x coordinate of the center of the block |
| YCORD | The y coordinate of the center of the block |

Table 2

ATTRIBUTES ASSOCIATED WITH THE ENTITY "NBD"

| A A A |
|--|
| ATTYINNTP |
| |
| And the owner of the owner own |

NBD NAME

N. SECTOR. CARS

N.ADJACENT

Description

The 4-character alphanumeric name associated with the neighborhood

The number of sector cars assigned to the neighborhood

The number of cars designated as adjacent resources for the neighborhood sector cars, but some neighborhoods may have no sector cars assigned to them.

Figure 1 illustrates the partition of a sample rectangular region into five neighborhoods assigned to three sector cars. (Note: The individual blocks constituting each neighborhood are not shown.) Neighborhoods 1 and 2 jointly constitute Sector A and Neighborhoods 2 and 3 constitute Sector B. Thus, Sectors A and B overlap. Sector CD is composed of Neighborhoods 4 and 5. One sector car is assigned to Sector A, one to Sector B, and the third is responsible for patrolling Sector CD. Patrol units are numbered from 1 to N.CAR (where N.CAR is the number of cars to be simulated) for internal reference. Table 3 describes the most important attributes associated with a patrol unit.

Sector cars and supervisory or special cars are differentiated only by their sector assignments and in the way they are nominated for dispatch. A sector car is assigned as the primary car for one or more neighborhoods and may be nominated as an alternate car for some other neighborhoods. This means that the simulation will attempt to assign it to jobs in its sector(s) if it is free. In addition, some types of jobs are only assigned to a sector car. Supervisory or special cars have no sector assignment and are dispatched only to high-priority jobs or to certain jobs when no sector cars are available.

MOVEMENT OF PATROL UNITS

To imitate the continuous movement of patrol units in the simulation would be very costly in terms of computing time and program complexity, and would change the results very little. Hence, the movement of patrol units is simulated as follows:

- grid. Distances and times are calculated as shown below.
- its next assignment.
- each car's sector is specified by the user.

• Patrol units move between points as if on a dense rectangular street • If a unit completes a job in its sector, it is placed "on patrol" at the location of the job just completed, where it remains until

• If a unit completes a job outside its sector and is not dispatched to another job, it returns to the centroid of its sector (traveling on the street grid) and goes "on patrol" there. The centroid of

-9-

5 4

Legend

| Sector | Skading | Neighborhoods Comprising Sector | Car Assigned |
|--------|---------|------------------------------------|-----------------|
| Α | | 1 and 2 | Α |
| В | | 2 and 3 | В |
| CD | | 4 and 5 | CD |

Fig. 1-An example of how neighborhoods and sectors are specified

| Attribute | |
|--------------|-------------------------------------|
| NUMB.SECTORS | The unit |
| CAR, NAME | A 4- the |
| XLOC | The of t |
| YLOC | The of t |
| ASSIGNMENT | The assi |
| CENTROID | The retu resp (usu cent |
| SPT | The parts |
| SOT | An ir = 1 i = 2 i = 3 i |
| SWT | An ir = 0 i = 1 i |
| | 1 = 2 i |
| | = 3 i o = 4 i |
| | s = 5 i |

-11-

Table 3

ATTRIBUTES ASSOCIATED WITH THE ENTITY "CAR"

Description

number of neighborhoods to which the is assigned as a sector car

-character alphanumeric name identifying patrol unit for output purposes

x coordinate of the current location he car

y coordinate of the current location he car

job to which the unit is currently gned (if any)

block to which the unit goes when it rns to patrol in its sector after conding to a job outside its sector ally chosen to be at or near the roid of the car's sector).

priority of the job currently being iced by the unit

ndicator variable that if the unit is working as primary car if the unit is working as backup car if the unit is working as tertiary car

ndicator variable that if the unit is on patrol in its sector if the unit is returning to sector rom an outside call if the unit is responding to a call in sector if the unit is responding to a call out of sector if the unit is working on a call in sector if the unit is working on a call out of sector = 6 if the unit is out of service

Simulating the movement of patrol cars in this way makes the probability that a car is patrolling a given block equal to the conditional probability of a call for service at that block given that there is a call for service in the sector. This way busy blocks receive more patrol than less busy ones. In some cases it may be desirable to specify different patrol frequencies, e.g., if a block has few calls for service but those that do occur are very serious. This can be done by modifying the routine PLACE.CAR.ON. PATROL (see page 40). One could provide the simulation with a set of block patrol frequencies for each car or for the region being simulated. When a car is to return to patrol, the block to which it returns would be sampled at random according to the specified probability distribution.

The model assumes that patrol units travel from point to point as follows: If a patrol unit is currently at Block 1 with coordinates (x_1, y_1) and is dispatched to a job at Block 2 with coordinates (x_2, y_2) , the simulation assumes that the unit travels parallel to the coordinate axes and that a pair of streets at right angles to each other pass through the points in question. Thus the program calculates the distance traveled as

$$d_{12} = |x_1 - x_2| + |y_1 - y_2|.$$

The time required to travel to Block 2 depends on the priority of the job, so that for a job of priority p we must specify a response velocity, v. The resulting response time is calculated as

$$t_{12} = d_{12}/v_p$$
.

The simulation assumes that grid coordinates are assigned so that one grid coordinate unit represents one mile in each coordinate direction. If this is not the case, the user should supply velocities in terms of grid units per hour (instead of miles per hour) so that response times can be expressed in terms of minutes.

THE JOB STREAM

Jobs are created or collected outside the simulation and placed on a file called the job stream, which is "read and played out" by the simulation much in the same way that a tape recorder reads and plays a tape of music. As we have already pointed out, this procedure has several advantages

over generation of jobs within the simulation. There are two types of jobs included in the job stream: (1) calls for service, and (2) out-of-service jobs. The calls for service represent jobs carried out by patrolmen as part of their police functions. The out-of-service jobs represent functions other than preventive patrol carried out by patrolmen during a normal tour of duty. These include taking meal breaks, getting gas, having minor repairs made to the patrol car, etc. The out-of-service jobs are always associated with a specific patrol car. Each call for service has the following information associated with

it in the job stream:

- enters the system.
- N.BLOCK) of the block at which the job occurs.
- this assumption in the model.)
- Table 4 describes the attributes associated with every temporary entity JOB, which represents a call for service. Each out-of-service job requires only two pieces of information:
 - of the car to be placed out of service. • Duration: the length of the out-of-service time.

The out-of-service jobs can be used to vary the number of patrol cars on duty over the course of the simulation. For example, if the calls for service represent calls over a whole day and if different numbers of cars

-13-

• Job entry time: 'The "simulated clock" time at which the job

• Priority: A number from 1 to 5 (1 is the highest priority). Job location: The internal identification number (from 1 to

• Job duration: The length of time that the longest working patrol car spends at the job. (It should be noted that in the real world it is possible for the seriousness and duration of an incident to depend on the speed with which a patrol car gets to the scene. Since the nature of this dependency is not known, the simulation assumes that job duration and seriousness are unaffected by the speed of response. For comparing many types of deployment changes-particularly where the changes in response times are small--this should not be a serious limitation, but the user should be aware of

• Patrol car: the internal identification number (from 1 to N.CAR)

patrol policies quite different from New York City practice can be simulated by modifying only the program's initialization data. For example, different numbers of cars can be sent to different kinds of incidents; sector boundaries can be shifted or sectors can be eliminated entirely; certain calls can be held for specific cars.

-15-

Some dispatching policies, however, can be simulated only by changing the dispatching event routine (JOB.ENTRY). For example, the dispatching routine presented here does not make decisions based on <u>actual</u> car locations. It references all incoming jobs to a neighborhood, and dispatch decisions are made by referencing an ordered list of patrol units for that neighborhood that has been read in at the beginning of the simulation. Another version of the dispatching routine that we have used references an incoming call to its exact location and uses the exact locations of all patrol cars to determine the closest available cars to an incident. This version has been used to evaluate the desirability of car location devices.

-14-

Table 4

ATTRIBUTES ASSOCIATED WITH THE ENTITY "JOB"

| <u>Attribute</u> | Description |
|------------------|--|
| PRIORITY | A number from 1 to 5 indicating the priority (seriousness) of the call |
| LOCATION | The internal reference number of the block at which the job occurs |
| DURATION | The length of time that the longest working unit spends on the job |
| STATUS | <pre>An indicator variable that = 0 if no cars have arrived = 1 if the primary car has arrived = 2 if some cars, but not the primary car, have arrived</pre> |
| ASSIGNED.CARS | A list of the patrol cars assigned to the job |

are on duty during each tour of the day, some cars could be put out of service for an entire tour at each tour-change time.

THE DISPATCHING ACTIVITY

The simulation imitates dispatching decisions, but does not simulate the dispatching activity per se. That is, it does not account for any time consumed during the dispatching function. For many applications, dispatching requires relatively little time and the dispatcher is not overloaded, so this model will be adequate. If, however, the dispatching-communication function is itself a potential source of delay, it would have to be modeled.

In the simulation, dispatching decisions are based on the location and priority of the job, and on the availability of patrol cars at the time of dispatch. A full description of how the simulation uses each of these characteristics in assigning cars is given in Section IV of this Report. The structure of the dispatching system we have coded is based on the computer-assisted dispatching system in use in New York City. We use the same type of information available to the New York City dispatcher and have rationalized the flexible rules and guidelines he follows. The simulation structure is general enough so that a variety of dispatching and

IV. EVENTS AND SUBROUTINES

The simulation program is composed of six event routines and the subroutines that support them. The events are:

- JOB.ENTRY A call for police service is received at the dispatching center.
- ARRIVAL.AT.SCENE A patrol unit arrives at the scene of an incident.
- CALL.END A patrol unit finishes work at an incident or other activity.
- RETURN.TO.SECTOR A unit that has responded to an incident out of its sector has returned to patrol in its own sector.
- OUT.OF.SERVICE It is time for some unit to go out of service (for a meal or another nonservice-related purpose). If the unit is free, it goes out of service; if it is busy, the out-of-service period begins later.
- END.OF.SIMULATION The simulation is over and statistics are printed out.

The progress of an incident can be traced through the system as follows (names in capital letters that are not event names are names of subroutines): At a certain time (specified as part of the input job stream), the call is received by the dispatcher, causing the JOB.ENTRY event to be executed. Associated with the call are its location, priority, and duration. The program uses the dispatch policy for the job's priority class (DISPR1, DISPR2, DISPR3, DISPR4, or DISPR5) to decide which car (or cars) to send. Each car to be dispatched is ASSIGNed to the job. In the ASSIGN routine its distance from the job is determined by executing CALCULATE.DISTANCE, and the car's ARRIVAL.AT.SCENE is scheduled by SCHEDULE.CAR.ARRIVAL. If a car to be sent to the job is traveling back to its sector, the simulation first carries out a CANCEL.SECTOR.RETURN, then determines the car's current position with INTERPOLATE.LOCATION, and, finally, ASSIGNs it to the job. If a car to be sent to the job is currently responding to a lower priority job, the simulation will PREEMPT it, INTERPOLATE, LOCATION, and ASSIGN it to the higher priority job.

Each ARRIVAL.AT.SCENE causes a CALL.END to be scheduled for the arriving car at a time that depends on whether the car is the primary car at the job or a backup or tertiary car. When a car finishes a job, the simulated dispatcher checks the queues for another job. If a job is waiting for this car, the simulation dispatches the car to it with REASSIGN.CAR.TO.JOB. If no job is waiting that can be assigned to it, the simulation will PLACE.CAR. ON.PATROL. If currently within its sector, the car remains where it is; otherwise, it proceeds back to the centroid of its sector. In the latter case, a RETURN.TO.SECTOR event is scheduled.

Meals and other occurrences that place a car out of service are treated in much the same way as jobs. A car may be scheduled for its meal at any time during a tour (see input data section). The car will then be automatically scheduled for a meal at this (relative) time during each tour. When a car is scheduled to go out of service, an OUT.OF.SERVICE event occurs. If the car is available (on patrol), it begins its out-of-service time immediately; otherwise, the event is queued as jobs are. As soon as the car next becomes available, it will be placed out of service. The completion of an out-of-service period is treated exactly as if the car were completing a job.

In Fig. 2 we illustrate the sequence of events and the order of subroutine calls in the simulation by presenting a flow chart for an example of a single Priority 1 call. Each outer box in the flow chart denotes a simulated event, with time increasing as one proceeds down one page and onto the next. Small boxes inside the outer boxes denote subroutines called by the event routines. The subroutines may call other subroutines, which are shown in still smaller included boxes. Each outer box is labeled with an identifier (e.g., Box 1, Box A(D)) that may be used to refer to that specific event elsewhere in the flow chart.

In the example being simulated, the arrival of the call for service is simulated by the exogenous event JOB.ENTRY, which reads the description of the call from the input job stream. This information includes the fact that it is a Priority 1 incident. It also includes the location of the incident and the length of time it will take for the primary car to service it.

The JOB.ENTRY event calls subroutine DISPR1 to choose the patrol cars that will be dispatched to this incident. It determines that four cars, named B, D, G, and F will be sent, with B acting as primary car, D as backup car, and G and F as tertiary cars. Each car is sent to the incident through a separate call to the subroutine ASSIGN, which performs some bookkeeping functions and calls SCHEDULE.CAR.ARRIVAL to schedule the ARRIVAL.

-16-

-17-





Schedule Box A(i)

►LOOP

Box 1

END. CALL. SCHEDULING Schedule Boxes E(D) and E(B) Car D(backup car) arrives first.

Car B, the primary car, arrives. Schedule end of call for all cars on the scene.



Fig. 2-Flow chart for Priority 1 incident

-18-

Fig. 2–(continued)



AT.SCENE event. The time required for each car to travel to the incident depends on the distance to be traveled and on the travel velocity. Car D is the first car to arrive. Car B, the primary car, arrives next. Its arrival allows the simulation to schedule CALL.ENDs for all cars already on the scene (in this case only car D). Cars G and F arrive subsequently and their CALL.ENDs are scheduled as soon as they arrive. Cars F and G, the tertiary cars, are the first to leave the scene. Car G, which has responded out of its sector, is placed on patrol and sent back to its sector (a RETURN.TO.SECTOR event is scheduled for it in Box E(G)). There is an out-of-service job waiting for car F, so it is placed out of service. When Car D, the backup car, is finished at the incident there is another job waiting for it. It is, therefore, dispatched to the new job. Car B, which is already in its sector, is placed on patrol at the scene of the incident after it completes its work there. The flow chart ends with the arrival of car G at the centroid of its sector where it remains on preventive patrol. Of course, in an actual simulation run, additional events associated with other incidents would be interspersed among the events shown.

A complete description of each of the event routines and their supporting subroutines is given below.

MAIN

Every SIMSCRIPT II program must contain a routine called "MAIN," On each simulation run, the program execution begins at the first instruction (line 32000 in our program) in this MAIN routine and proceeds from there. In MAIN, the computer reads in the initialization deck, sats the dimensions for all arrays, and sets initial values for variables and arrays. It also prints a report displaying most of the initialization data. Before passing control to the internal SIMSCRIPT timing routine that controls the execution of the events, the program schedules the first meal time for each of the patrol cars, and then schedules the end of the simulation. The timing routine is part of the SIMSCRIPT compiler, and so is not included in our program documentation. The MAIN routine is listed below.

-21-

MAIN 00031900 00032000 DEFINE SIM. LENGTH AND MEAL. TIME AS REAL VARIABLES 00032100 RESERVE REGION.NAME (*) AND TITLE (*) AS 20 00032200 START NEW PAGE FOR I = 1 TO 20, READ TITLE(1) AS (20) A 4 00032300 FOR I = 1 TO 20, READ REGION.NAME(I) AS (20) A 4 00032400 READ SIM.LENGTH, MAX.SENT, MEAL-DURATION, AND TOUR.LENGTH 00032500 WRITE AS " INPUT DATA FOR " 00032600 00032630 FOR I= 1 TO 20, WRITE REGION.NAME(I) AS 00032660 WRITE AS / 00032700 SKIP 3 LINES PRINT 1 LINE WITH SIM.LENGTH AS FOLLOWS 00032800 SIMULATION LENGTH =****** . ** HOURS 00032900 00033000 SKIP I LINE 00033100 READ N. BLOCK + N. NBD + N. CAR 00033200 LET N. AVAILABLE=N. CAR PRINT 1 LINE WITH N. BLOCK . N. NBD AND N. CAR AS FOLLOWS 00033300 NO. OF BLOCKS = **** NO. OF NBDS. = **** NO. OF CARS = **** 00033400 RESERVE ADJACENT.CARS (*+*) AS N.NBD BY N.CAR+AND SECTORS (*+*) 00033500 AS N.CAR BY # 00033600 CREATE EVERY BLOCK, NBD, CAR, AND P.CLASS(5) 00033700 SKIP 1 OUTPUT LINE 00033800 PRINT 1 LINE AS FOLLOWS 00033900 BLOCK ID TAX NO. NOD ID 00034000 FOR I=1 TO N.BLOCK, DO 00034100 READ B.TAXNO(B) .NBDID(B) .XCORD(B) .YCORD(B) USING UNIT 7 00034200 PRINT 1 LINE WITH B, TAXNO(B), NBDID(B), XCORD(B), YCORD(B) THUS 00034300 *** 6666 *** ******* 00034400 LOOP 00034500 START NEW PAGE 00034600 FOR I= 1 TO 20 + WRITE TITLE(I) AS 00034700 A 4 WRITE AS / 00034750 SKIP 2 LINES 00034800 PRINT 2 LINES AS FOLLOWS 00034900 NBD ID SECTORS NUMBER OF ORDER IN WHICH 00035000 SECT.CARS ADJ.CARS CARS ARE NOMINATED 00035100 FOR I=1 TO N.NBD. DO 00035200 READ NONBDONAME(N) ON SECTOR CARS(N) ON ADJACENT(N) 00035300 FOR J=1 TO N.CAR.READ ADJACENT.CARS(N.J) 00035400 PRINT 1 LINE WITH NONBDONAME (N) ON SECTOR CARS (N). 00035500 N. ADJACENT (N) AS FOLLOWS 00035600 *** *** *** 00035700 FOR K=1 TO N.CAR.DO PRINT I LINE WITH 00035800 ADJACENT. CARS(I,K) AS FOLLOWS 00035900 ** 00036000 LOOP 00036100 LOOP 00036200 FOR I=1 TO N.CAR 00036300 READ 'C+CAR.NAME (C) .NUMB.SECTORS (C) .CENTROID (C) .MEAL.TIME 00036400 RESERVE SECTORS (C+*) AS NUMB. SECTORS (C) 00036500 FOR J=1 10 NUMB.SECTORS(C). READ SECTORS(C,J) 00036600 LET XLOC(C)=XCORD(CENTROID(C)) 00036700 LET YLOC(C)=YCORD(CENTROID(C)) 00036800 SCHEDULE AN OUT, OF SERVICE GIVEN C IN MEAL, TIME HOURS 00036900 LOOP 00037000 FOR I=1 TO N.P.CLASS READ VELOCITY(I) 00037100 RESERVE P.DURATION(+) AS 3 00037200 FOR T = 1 TO 3 READ P.DURATION(I) 00037300 CREATE AN END. OF. SIMULATION 00037400 SCHEDULE THIS END. OF. SIMULATION IN SIM. LENGTH HOURS 00037500 START SIMULATION 00037600 STOP 00037700

00037800

END

EVENT FOR JOB.ENTRY

This external event represents the arrival of a new incident into the system. For most types of jobs (all priorities except 4), the job arrives at the dispatching center. It is presumed that at this point the dispatcher has already obtained all information necessary for dispatch and can immediately assign a patrol car if one is available. If there are delays due to the dispatching operation itself, it will be necessary to have an event or events that can account for the dispatching delays. This might involve explicit modeling of the dispatcher's operation, including the modeling of delays that may occur when the citizen reporting the incident attempts to make telephone contact with the police, or while the police are obtaining the relevant information from the caller. We describe the simulated dispatching operation below in general terms. Note that Priority 4 jobs are intended to model those incidents that are discovered by the patrolling units themselves and that are not first reported to the police by telephone. Patrol cars are not, of course, dispatched to these jobs. Rather, the job is "picked up" by the appropriate unit. Yet in the following discussion we use the term "dispatch" somewhat loosely and incorporate under it the assignment of cars to Priority 4 jobs. And as a modeling convenience, we also place pickup jobs in a fictional queue where they "wait" for the appropriate patrol car to pick them up.

Data describing the new incident--its location, priority, and duration--are included as part of the input job stream. In order to determine whether there are patrol resources available to dispatch to the call and, if so, which resources to dispatch, the JOB.ENTRY event routine calls the appropriate dispatch subroutine based on the priority of the call for service. The dispatch subroutines in the version of the program documented here are called DISPR1, DISPR2, DISPR3, DISPR4, and DISPR5, corresponding to the five priority classes. The dispatch policy for each priority class is given in the description of each dispatch subroutine.

A general flow chart for the JOB.ENTRY event, including all the subroutines called by the various dispatch subroutines, is given in Fig. 3.

-23-



EVENT FOR JOB ENTRY DEFINE HOUR AND MOUR AS REAL VARIABLES READ LOC. IPR. HOUR AND MOUR CREATE A JOB CALLED J LET ENTRY. FIME (J) = TIME . V LET LOCATION (J)=LOC LET PRIORITY(J) = IPR LET DURATION(J) =HDUR*60.0 + MDUR LET STATUS(J)=0 GO TO PR(IPR) IPR(1)1 CALL DISPRI(J) RETURN ·PR(2) CALL DISPRZ(J) RETURN 1PR(3)1 CALL DISPR3(J) RETURN 1PR(4)1 CALL DISPR4(J) RETURN PR(5) . CALL DISPR5(J) RETURN END

Dispatch Subroutines: DISPR1, DISPR2, DISPR3, DISPR4, DISPR5

These subroutines are the heart of the simulation. With its five dispatching procedures, the simulation is capable of testing a wide range of dispatching alternatives by changing only the priorities assigned to jobs or the sector assignments of the patrol cars. If the user would like to test a policy that cannot be handled in this way, he can write his own dispatching routine(s) and substitute it for one or more of the five discussed here.

In all five policies the incoming jobs are referenced to a neighborhood, and dispatch decisions are made using an ordered list of patrol units associated with that neighborhood. The list consists of all patrol units that are to be considered possible candidates to be assigned to a job located in the neighborhood in question. Whether the unit is assigned, and in what capacity it will be assigned, depends upon the current status of the unit and of other units, upon the priority of the job, and upon the values given to various dispatch parameters. (Since neighborhoods can be as small as a single block or as large as the entire region being simulated, great flexibility is possible in the dispatching strategies that can be simulated.)

The following discussion of each of the dispatching procedures in the simulation depends on an understanding of the ordered list of patrol units. There is one such list for every neighborhood. For a particular neighborhood, say I, it is called ADJACENT.CARS(I). At the head of the list is the sector car(s) for the neighborhood, next come the sector cars for adjoining neighborhoods, then other cars in the region, and finally supervisory or

| | ^ | - | |
|---|---|---|--|
| - | Z | C | |

| 00059800 |
|----------|
| 00059900 |
| 00060000 |
| 00060100 |
| 00060200 |
| 00060300 |
| 00060400 |
| 00060500 |
| 00060600 |
| 00060700 |
| 00060800 |
| 00060900 |
| 00061000 |
| 00061100 |
| 00061200 |
| 00061300 |
| |

special cars, if there are any. As will be seen in the following discussion, sector cars, neighboring sector cars, etc., are treated differently in dispatching. Associated with the list ADJACENT.CARS(I) are two pointers that are used in the dispatching routines:

- N.SECTOR.CARS(I) tells how many sector cars are assigned to Neighborhood I.
- N.ADJACENT(I) tell how many "adjacent sector cars" are associated with Neighborhood I.

Figure 4 is a representation of the ordered list for Neighborhood I and its members. We assume for illustration that there are six patrol units in the field, numbered 1 through 6, and one supervisor's car. numbered 7. There is one sector car for Neighborhood I, namely car 4; and there are two adjacent cars, numbered 3 and 1. Thus, N.SECTOR.CARS(I) = 1 and N, ADJACENT(I) = 2, and as shown schematically in Fig. 3 these point to appropriate cars on the list.

| | ADJACENT, CARS(1) | |
|-----------|-------------------|-------------------------|
| | car 4 | N.SECTOR, CARS(I) = 1 |
| | car 3 | |
| | car 1 | N.ADJACENT(I) = 2 |
| | car 2 | |
| | car 5 | |
| · · · · · | car 6 | |
| | car 7 | |

Fig. 4-The structure of the dispatcher's list ADJACENT.CARS(I)

When a job in Neighborhood I comes into the system, the car(s) dispatched depends on the priority of the job and on the ordered list ADJACENT. CARS(I). A description of each of the five dispatch subroutines follows. As mentioned above, each of the subroutines calls the ASSIGN routine to assign a car to a job. If a car to be sent to a job is returning to its sector, the routines CANCEL.SECTOR.RETURN and INTERPOLATE.LOCATION are called before the ASSIGN routine is executed.

ROUTINE TO DISPRI GIVEN J. Priority 1 jobs are the most serious incidents. They generally include robberies in progress, police officers in

danger, etc. In DISPR1, the simulation dispatches the first MAX.SENT of the available cars on ADJACENT.CARS(I). * If fewer than MAX.SENT are available, all of the available cars are dispatched. Those cars dispatched to the job are all associated with the internal identification number assigned to the job (variable J). Cars are available if they are on patrol or responding to lower priority incidents. The preempting of cars that are responding to lower priority calls in order to respond to Priority 1 jobs is the only preempting included in the simulation. Starting from the top of ADJACENT.CARS(I), the available cars are assigned as follows:

- A primary car (the first available car on the list)--the job is the entire duration.
- A backup car (the second available car on the list)--works for a upon the arrival of the primary car.
- Tertiary cars (all other available cars on the list up to a total the primary car.

If no cars are available it is assumed that the job is handled by units from another region. Where this occurs, it is recorded by adding "1" to the variable OUTSIDE.DISPATCH and the job is not considered further by the simulation. The number of times this happens is shown in the output report "Number of Precinct Cars Sent to Calls" (Fig. 16). Thus, the simulation never queues Priority 1 jobs.

ROUTINE TO DISPRI GIVEN J LET SECTENBDID(LC ATION(J)) LET MARKER=0 FOR J=1 TO N. CAR, DO .. NOMINATE CARS .. LET K=ADJACENT. CARS(SECT.I) IF SWT(K) LE 3 AND SPT(K) NE 1 ** CAR IS ELIGIBLE ** IF MARKER=0 LET SOT(K)=1 LET MARKER=1 GO TO CALCI OTHFRWISE IF MARKER = 1 LET SOT(K)=2 LET MARKER=2 GO TO CALCI

*MAX.SENT is a variable whose value is specified by the user. It must be less than or equal to N.CAR.

-26-

-27-

assumed to start upon its arrival at the scene and it works for

proportion, P.DURATION(2), of the duration of the job, starting work

of MAX.SENT-2)--each of these work for a proportion, P.DURATION(3), of the duration of the job, starting work upon the arrival of

| OTHEOWISE TE MARKER = MAX.SENT GO OUT | 00062500 |
|---|----------|
| OTHERWISE IN ANDRED LET SOT (K) #3 | 00062600 |
| UTHERWISE ADD I TO MARKER ELC SUVEN K AND ASSIGNMENT (K) | 00062700 |
| (CALCI IF SWT(K)=1 CALL INTERPOLATE COLATION GIVEN & AND HOSTOLE | 00062800 |
| REGARDLESS | 00062900 |
| IF SPT(K) GT 1 CALL PREEMPT(K) | 00063000 |
| REGARDLESS CALL ASSIGN(K,J) LEI SWI(K)=2 | 00063100 |
| IF I GT N.SECTOR.CARS(SECT) LET SWT(R)=3 | 00063300 |
| REGARDLESS | 00063200 |
| REGARDLESS | 00063300 |
| | 00063400 |
| THE WARKER TO ADD 1 TO OUTSIDE DISPATCH | 00063500 |
| TOUT IF MARKER - O ADD I DESTROY JOB CALLED J | 00063600 |
| | 00063700 |
| REGARDLESS LET CUONTIT/SEARCE | 00063800 |
| RETURN | 00063900 |
| END A CONTRACT OF | |

ROUTINE TO DISPR2 GIVEN J. Priority 2 jobs are less serious than Priority 1 jobs. Either one or two patrol cars are assigned to Priority 2 jobs, as follows:

- If no cars are available in the region being simulated, the job is queued for the first available car. Jobs in the Priority 2 queue get assigned units on a first-in-first-out basis before jobs in other queues. A job that is queued gets assigned only one car.
- One car is sent unless a sector car is available to act as primary car and another sector car or one of the N.ADJACENT cars from the neighboring sectors is available. In this case, both a primary and backup car are sent. The primary car works for the duration of the job and the backup car works for a proportion, P.DURATION(2), of the duration. Work starts upon the arrival of the primary car.

| POULTINE TO DISPRO GIVEN J | 00064000 |
|--|----------|
| ET SECT-NEDID () OCATION ()) LET MARKER=0 | 00064100 |
| LET DECINOLOGICO TINOMINATE CARS 11 | 00064200 |
| FOR THE TO HUGHNY DO CARS (SECT I) | 00064300 |
| TE OPT (K) =0 II CAR IS FITGIRIE II | 00064400 |
| IF MARKER=0 IFT SOT(K)=1 LET MARKER=1 GO TO CALC2 | 00064500 |
| | 00064600 |
| TE MADE TAND T LE N. AD MCENT (SECT) +N. SECTOR. CARS (SECT) | 00064700 |
| IF MARKENAL AND I LE NEMOROACH (SENT) STOLET OF THE | 00064800 |
| TO A DATE AND A DATE A | 00064900 |
| IF SMIN/AL CALL INTER STATES OF A STATES | 00065000 |
| DEGADDLESS CALL ASSTON(K.I) LET SWT(K)=2 | 00065100 |
| REGARDLESS CALL ASSISTOR OF LET SWI(K)=3 | 00065200 |
| TE A DI NOSCIONOCAUSTOLOTY EL SUTURY O | 00065300 |
| | 00065400 |
| | 00065500 |
| REGARDLESS | 00065600 |
| | 00065700 |
| THE MARKED FILL O IN GOLDLET ALTONN | 00065800 |
| UINERWIJE LEI LUUNIIC/AMARKEN REIUNN | 00065900 |
| na se d e mo les en la constante de | |

ROUTINE TO DISPR3 GIVEN J. Exactly one car is assigned to Priority 3 jobs. This car can be any available car in the region being simulated. The car dispatched is the first available car on the ADJACENT.CARS list associated with the neighborhood from which the call emanated. If no cars in the region are available the job is queued for the first available car, Priority 3 jobs are dispatched only if the Priority 2 queue (QUEUE(2)) is empty.

ROUTINE TO DISPR3 GIVEN J .. SEND ANY AVAILABLE CAR .. LET SECT=NBDID(LOCATION(J)) FOR I=1 TO N.CAR . DO LET K=ADJACENT.CARS(SECT.I) IF SPT(K)=0 **IICAR IS ELIGIBLE!!** LET SOT (K) =1 IF SWT(K)=1 CALL INTERPOLATE.LOCATION GIVEN K AND ASSIGNMENT(K) REGARDLESS CALL ASSIGN(K, J) LET SWT(K)=2 IF I GT N.SECTOR.CARS(SECT) LET SWT(K)=3 REGARDLESS LET COUNT(3)=1 RETURN OTHERWISE LOOP FILE J IN QUEUE(3) RETURN END

ROUTINE TO DISPR4 GIVEN J. These jobs represent "pickup" jobs--incidents If the appropriate sector car is available when the Priority 4 job is

spotted by patrolmen during the course of their preventive patrol activities. As is the case for Priority 3 jobs, exactly one car is assigned. But the car assigned will always be a sector car whose area of responsibility includes the block where this incident occurs. Since the car is already at the location of the incident when it occurs, it does not have to travel to reach the scene. To imitate this, the simulation instantaneously changes the car's location to the grid coordinates associated with the pickup job. received, it will be immediately assigned to the job. If no sector car is available, there are two possible courses of action: (1) the call is queued until a sector car is available or (2) the call is ignored (since no car was available at that instant to spot the incident). The former approach is taken in this simulation program. This approach is appropriate when the calls for service in the job stream represent an actual historical sequence of calls. In this case, the fact that a specific car was unavailable when a pickup job occurred indicates a difference between the real and simulated worlds. But elimination of the job would result in

-28-

changing the workload of the cars on duty and would lead to difficulties in comparing the simulation results produced for the same job stream by different deployment policies.

However, there is an alternative way that pickup jobs can be inserted into the job stream. They may be generated randomly according to the (known) frequency with which a car on preventive patrol in a region would encounter pickup jobs. In this case, if a Priority 4 call arrives when the sector car is busy, the call should be ignored. As a result, the more time a car has for preventive patrol, the more often it will service pickup jobs. Similarly, a busy car will have correspondingly fewer pickup jobs. In order to modify the simulation to implement this approach to Priority 4 jobs, the following statement should be substituted for statement number 00069100 in ROUTINE TO DISPR4:

DESTROY JOB CALLED J.

(Even in this case there will be difficulties in comparing the simulation results produced for the same job stream by different deployment policies.)

| ROUTINE TO DISPR4 GIVEN J | 00067700 |
|--|----------|
| NA PICK-UP JOB NA | 00067800 |
| LET SECT = NBDID (LOCATION (J)) | 00067900 |
| FOR J = 1 TO N.SECTOR.CARS(SECT), DO | 00068000 |
| LET K = ADJACENT.CARS(SECT.I) | 00068100 |
| IF SPT(K) = 0 + CAR IS AVAILABLE + + | 00068200 |
| LET XLOC(K) = XCORD(LOCATION(J)) | 00068300 |
| LET YLOC(K) = YCORD(LOCATION(J)) | 00068400 |
| CALL ASSIGN(K,J) | 00068500 |
| LET SWT(K) = 2 LET SOT(K) = 1 | 00068600 |
| LET COUNT(4) = 1 | 00068700 |
| RETURN | 00068800 |
| OTHERWISE | 00068900 |
| | 00069000 |
| FILE J IN QUEUE(4) PIND SECTOR CAR FREEPP | 00069100 |
| RETURN TO THE ADDRESS OF THE ADDRESS | 00069200 |
| END | 00069300 |
| | 00069400 |

ROUTINE TO DISPR5 GIVEN J. Priority 5 jobs are similar to Priority 4 jobs, but no instantaneous travel is assumed. These jobs are assigned only to a sector car for the neighborhood associated with the job. This class is meant to imitate low priority jobs that may be assigned by a supervisor, such as taking reports on past burglaries, etc. If no sector car is available the job is placed on QUEUE(5). A Priority 5 job is assigned only when the Priority 2 and Priority 3 queues are empty and QUEUE(4) does not contain a job waiting for the same sector car(s).

ROUTINE TO DISPRS GIVEN J IT ASSIGN JOB ONLY TO SECTOR CAR IT LET SECTENBDID (LOCATION (J)) FOR I= 1 TO N.SECTOR.CARS(SECT) + DO "LOOK FOR A SECTOR CAR!! LET K#ADJACENT.CARS(SECT.I) TF SPT(K)=0 MCAR IS ELIGIBLE .. IF SWT(K)=) CALL INTERPOLATE.LOCATION GIVEN K AND ASSIGNMENT (K) REGARDLESS CALL ASSIGN(K.J) LET SWT(K)=2 LET SOT(K)=1 LET COUNT (5) #1 RETURN OTHERWISE LOOP FILE J IN QUEUE(5) **NO SECTOR CAR FREE** RETURN END

ROUTINE TO INTERPOLATE.LOCATION GIVEN K AND J

This routine is called by each of the dispatch subroutines whenever the car to be dispatched is en route to a different location (either returning to its sector from a previous job or responding to a lower priority job). It calculates the approximate current position of CAR K, which is traveling to JOB J (the "job" might be a return to patrol duty), and assigns the x and y coordinates of CAR K's current simulated position to the variables XLOC(K) and YLOC(K). The routine makes the assumption that a car travels only in right-angle directions during the course of a trip--first traveling on a straight path in the x direction, then on a straight path in the y direction. The total distance already traveled by the car is assumed to be equal to the fraction of the total duration of the trip that has already elapsed.

If CAR K had been on its way to its sector, the event RETURN.TO.SECTOR, which had already been scheduled for this car, is now canceled by calling the CANCEL.SECTOR.RETURN routine.

ROUTINE TO INTERPOLATE LOCATION GIVEN K AND J. ** CALLED FROM PREEMPT AND FROM JOB.ENTRY ** NORMALLY MODE IS REAL DEFINE LJ.K.J.R.PATROL.RETURN AS INTEGER VARIABLES LET LJ=LOCATION(J) LET V=VELOCITY(PRIORITY(J)) LET XD=XCORD(LJ)-XLOC(K) LET YD=YCORD(LJ) - YLOC(K) LET T=(TIME.V - T.START.RESPONSE(K))*24 LET TX=ABS.F(XD)/V LET DELT=T - TX IF DELT GT O LET XLOC(K) = XCORD(LJ) LET YLOC(K) = V + DELT + SIGN + F(YD) + YLOC(K) GO TO CHECK . IF . PATROL OTHERWISE LET XLOC(K) =V# (=DELT) #SIGN F (XD) +XLOC(K)

-30-

| ICHECK, IE . PATROL | TE SWT (K) NE 1 | RETURN | 00072600 |
|---------------------|-----------------|------------------------------|----------|
| | OTHERWISE CALL | CANCEL SECTOR RETURN GIVEN K | 00072700 |
| | RETURN | | 00072800 |
| C110 | | | 00072900 |

-32-

END

ROUTINE TO CANCEL.SECTOR.RETURN GIVEN K

This subroutine is called when CAR K, which is returning to its sector, is dispatched to a new job before it reaches its sector. It cancels the event RETURN.TO.SECTOR, which had already been scheduled for this car.

| ROUTINE TO CANCEL.SECTOR.RETURN GIVEN K | 00074000 |
|---|----------|
| LET PATROL .RETURN = ASSIGNMENT(K) | 00074100 |
| DESTROY JOB CALLED PATROL.RETURN | 00074200 |
| LET R=NEXT.EVENT(K) | 00074300 |
| CANCEL RETURN. TO SECTOR CALLED R | 00074400 |
| DESTROY RETURN.TO.SECTOR CALLED R | 00074500 |
| RETURN | 00074600 |
| END | 00074700 |

ROUTINE TO PREEMPT(K)

If a car is responding to a lower priority incident when the dispatcher receives a Priority 1 call, this routine redirects it to the Priority 1 incident. It cancels CAR K's ARRIVAL.AT.SCENE for the lower priority incident and removes it from the list of ASSIGNED. CARS for that incident.

If no car has yet arrived at the lower priority incident, it places the incident on the appropriate QUEUE for redispatch. Otherwise, it readjusts the assignments of the cars at the lower priority incident and, if the primary car is the one that has been preempted, it reschedules the end of the incident.

| an bar an | 00074800 |
|--|-----------|
| ROUTINE TO PREEMPT(K) | 00074900 |
| IET J = ASSIGNMENT(K) | 00075000 |
| TE J IS NOT IN A QUEUE | 00075100 |
| IF STATUS(J)=0 FILE J FIRST IN QUEUE(PRIORITY(J)) | 00075200 |
| REGARDLESS | 00075300 |
| REGARDLESS LET E= NEXT. EVENT(K) CANCEL THE ARRIVAL.AT. SCENE CALLED F | 00075400 |
| DESTROY THE ARRIVAL.AT. SCENE CALLED E | 00075500 |
| REMOVE & FROM ASSIGNED. CARS(J) LET SPT(K)=0 | 00075600 |
| NOW INTERPOLATE LOCATION GIVEN K AND J | 00075700 |
| IF N.ASSIGNED.CARS(J) =1 LET F=F.ASSIGNED.CARS(J) | 00.075800 |
| IF SOT(5)=2 AND SWT(F) GE 4 | 00075900 |
| LET E 🕊 NEXT.EVENT(F) | 00076000 |
| CANCEL THE CALL.END CALLED E | 00076100 |
| RESCHEDULE THE CALL.END CALLED F. IN DURATION (J) MINUTES | 00076200 |
| REGARDLESS | 00076300 |
| LET SOT(F) *1 | 00076400 |
| REGARDLESS | 00076500 |
| LET T.START.RESPONSE(K) = TIME.V | 00076600 |
| RETURN | 00076700 |
| END | 00076800 |
| | |

ROUTINE TO ASSIGN(K, J)

This routine assigns CAR K to work on JOB J. It assigns the car the priority associated with the job and places the car on the list of cars assigned to this job (ASSIGNED.CARS(J)). It also schedules the time at which the car will arrive at the scene by calling the routine SCHEDULE.CAR. ARRIVAL(K,J).

```
ROUTINE TO ASSIGN (K, J)
LET SPT(K)=PRIORITY(J)
LET ASSIGNMENT(K)=J
WOW SCHEDULE . CAR . ARRIVAL (K . J)
FILE K IN ASSIGNED.CARS(J)
   IF SWT(K) LE 1 SUBTRACT 1 FROM N.AVAILABLE REGARDLESS
RETURN
END
```

ROUTINE TO SCHEDULE CAR. ARRIVAL (K, J)

This routine, called from the ASSIGN subroutine, schedules the arrival of CAR K at the location of JOB J. The function CALCULATE.DISTANCE is used to calculate the distance to be traveled. It is assumed that the car travels from its current location at a constant velocity that is determined by the priority of the job. The event that ends the trip is ARRIVAL.AT.SCENE. The identity of this event is stored in the attribute NEXT. EVENT(K), and the current time is stored in T.START.RESPONSE(K).

```
ROUTINE TO SCHEDULE.CAR.ARRIVAL(K&J)
DEFINE DIST AS A REAL VARIABLE
LET DIST=CALCULATE.DISTANCE(XCORD(LOCATION(J)),YCORD(LOCATION(J)),
XLOC(K) , YLOC(K))
LET T.START.RESPONSE(K)=TIME.V
CREATE AN ARRIVAL.AT.SCENE
SCHEDULE THIS ARRIVAL.AT. SCENE GIVEN K IN
DIST/VELOCITY (PRIORITY (J)) HOURS
LET NEXT. EVENT (K) = ARRIVAL. AT. SCENE
RETURN
END
```

ROUTINE TO CALCULATE. DISTANCE GIVEN X1, Y1, X2 AND Y2

This routine is used to calculate the distance, D, between a point with grid coordinates (X1, Y1) and a point with grid coordinates (X2, Y2). The function may be easily changed to suit the city being simulated. The version of the routine that is reproduced here calculates the right angle distance, as follows:

00077100 00077200 00077300 00077400 00077500 00077600 00077700 00077800 00077900 00078000

00077000

D = | X2 - X1 | + | Y2 - Y1 |.

This equation assumes that a dense rectangular street network exists, connecting all pairs of points. For some of our experiments we found it necessary--and easy--to modify the function to model the effects of natural or man-made barriers such as large bodies of water or railroad tracks.

-34-

| ROUTINE TO CALCULATE DI | STANCE GIVEN X1.Y1.X2 Y2 AS REAL VARIABLES | AND YS | 0008240 |
|----------------------------|---|--------|---------|
| LET D = ARS.F($X2-X1$) + | ABS.F (Y2-Y1) | | 0008260 |
| END | | | 009590 |

EVENT FOR ARRIVAL.AT.SCENE(K)

*

This internal event, which is flow-charted in Fig. 5, represents the arrival of CAR K at the scene of an incident. The first statements in the routine change the state of the system to reflect this event. What follows depends on whether the car has been assigned to be the primary car for the incident or one of the backup cars. If it is the primary car and no other car has arrived, its CALL.END event is scheduled. If other cars have arrived, CALL.END events are scheduled for each of these cars (since the duration of a job is measured from the time of arrival of the primary car). If the car is a backup car and the primary car is already there, then a CALL.END is scheduled for the car. Otherwise, it must wait for the arrival of the primary car before its CALL.END can be scheduled. Finally, if this is the first car to arrive at the scene, its travel time is calculated and added to both the overall travel time statistics and the travel time statistics corresponding to the priority of this job.

| EVENT FOR AN | RIVAL .AT. SCENE (K) | | | 00078200 |
|---------------|---------------------------|--------------------|-------------------|-----------|
| DEFINE D AS | A REAL VARIABLE | | | 00078300 |
| LET LKARELOC | ATION (ASSIGNMENT (K) |) | | 00078400 |
| LET XLOC(K) = | XCOHD (LKAR) | | | 00078500 |
| LET YLOC(K) = | YCORD (LKAR) | | | 00078600 |
| IF SWT (K)=2 | LET SWT (K) =4 ADD | 1 TO IN.SECT.JOBS(| K) GO TO BOTH | 00078700 |
| OTHERWISE | LET SWT (K) #5 ADD | I TO OUT.SECT.JOBS | (K) | 00078800 |
| +BOTH+ | | | | 00078900 |
| LET JEASSIGN | MENT (K) | | | 00079000 |
| LET JPR=PRIO | RITY (J) | | | 0007910 |
| LET D = LURA | TION (J) | | | 00079200 |
| TF SOT(K)=1 | ITHIS IS THE PRIMA | RY CAR AT JOP JII | GO TO PRIME.CAR | 0007930 |
| OTHERWISE | | | | 0007940 |
| IF STATUS(J) | =0 ++NO CAR HAS AR | RIVED AND THIS CAR | IS NOT PRIMARY | 00079500 |
| GO TO FIRST. | CAR | | | 0007960 |
| OTHERWISE | | | | 0007970 |
| IF STATUS(J) | =1 ++ PRIMARY CAR | HAS ALREADY ARRIVE | D II | 0007980 |
| PERFORM END. | CALL.SCHEDULING GIV | EN K AND D | | 0007990 |
| REGARDLESS P | ETURN IN IF NO PRIM | ARY CAR DELAY SCHE | DULING & CALL.END | 100080000 |
| FIRST.CAR! | | | | 00080100 |
| | | | | |





| LET STATUS(J)=2 | 002080200 |
|--|-----------|
| ET JOR PESPONSE TIME (JPR) = (TIME V = | 00080300 |
| T. START, RESPONSE (K)) #1440.0 | 00080400 |
| GO TO BT (JPR) | 00080500 |
| +RT(1) + LET P1.NBD.RSPOMSE(NUDID(LKAR))=J0H.RESPONSE.TIME(JPR) RFTURN | 00080600 |
| +RT(2) + LET P2.NBD.RSPONSE(NBDID(LKAR))=J08.RESPONSE.TIME(JPR) RETURN | 00080700 |
| +RT(3) + LFT P3.NBD.RSPONSE(NBDID(LKAR))=J08.RESPONSE.TIME(JPR) RETURN | 00080800 |
| +RT(4) + LFT P4.NBD.RSPONSE(NBDID(LKAR))=JOB.RESPONSF.TIME(JPR) RETURN | 00080900 |
| +RT(5) + LET PS.NBD.RSPONSE(NHDID(LKAR))=JOH.RESPONSE.TIME(JPR) RETURN | 00081000 |
| | 00081100 |
| A CTATIC / IN NE A SE COME CADE ALDEADY THEDESS | 00081200 |
| IF STATUSTUS NE V V SUME GARS REPEAULT THERE . | 00081300 |
| PUR CALIN LAR IN ASSIGNED CANSION WITH SWITCAN DE + DU | 00081400 |
| PERFORM ENDICALLISTREDULING GIVEN CAR AND D | 00081500 |
| LOOP LEI STATUS(J)=1 RETURN | 00081600 |
| OTHERMISE | 00081700 |
| LF1 JUH-RESPUNSE I IME (JPR) = (IIMC • V = | 00081800 |
| 1. START. RESPONSE (K)) *1440.0 | 00081900 |
| CALL END.CALL.SCHEDULING GIVEN R AND D | 00093000 |
| LET STATUS(J)=1 | 00002000 |
| GO TO RT (JPR) | 00002100 |
| END | 00006200 |

-36-

ROUTINE FOR END. CALL. SCHEDULING GIVEN K AND DUR

This routine, called from the ARRIVAL.AT.SCENE event, schedules the CALL.END event for CAR K. DUR is the length of time that the primary car will spend at the incident (i.e., the length of the incident). T, the length of time CAR K spends at the incident, depends on whether it is the parimary car (SOT(K)=1), a backup car (SOT(K)=2), or a tertiary car (SOT(K)=3), as follows:

T = P.DURATION(SOT(K)) * DUR,

where P.DURATION(I) is the proportion of DUR that a car of type I spends at an incident. The CALL.END for CAR K is then scheduled to occur in T minutes.

| | | 00082900 |
|--------|--|----------|
| | FOR END. CALL SCHEDULING GIVEN & AND DUR | 00083000 |
| DEETNE | AND DUR AS REAL VARIABLES | 00083100 |
| IFT T. | P. DURATION (SOT (K)) *DUR | 00083200 |
| CREATE | A CALL FND | 00083300 |
| SCHEDI | IF THIS CALL FND GIVEN K IN T MINUTES | 00083400 |
| RETURN | | 00083500 |
| ENR | | 00083600 |

EVENT CALL.END GIVEN K

4

This internal event is called whenever a car is scheduled to finish work at an incident or to return to service after being out of service. The subroutine first adjusts the state of the system and checks whether any other cars are still assigned to the incident. If there are none, the incident is erased from the system. Next, a check is made to see if this car has



been scheduled to go out of service. If so, the car is placed out of service for the required amount of time. Otherwise, the QUEUE is checked (from highest priority to lowest) to see if this car is needed at, and eligible for, some other incident. If so, REASSIGN.CAR.TO.JOB is called to dispatch the car to that incident. If there is no job waiting for this car, PLACE. CAR.ON.PATROL is called to return the car to preventive patrol in its sector. A flow chart for this event is given in Fig. 6.

EVENT CALL FND GIVEN K 00083700 IF SWT (K) =6 GO TO CHECK.Q ITEND OF OUT-OF-SERVICEII 00083800 OTHERWISE LET J=ASSIGNMENT(K) 00083900 REMOVE K FROM ASSIGNED. CAPS(J) 00084000 TF ASSIGNED CARS(J) IS EMPTY DESTROY JOB CALLED J 00084100 00084200 PEGARDLESS CHECK .Q. 00084300 FOR EACH JOB ON QUEUE(1) "THIS IS THE OUT-OF-SERVICE QUEUE !! 00084400 WITH LOCATION (JOB) *K. DO LET ASSIGNMENT (K) = JOB LET SWT (K) = 6 00084500 SCHEDULE & CALL-END GIVEN K IN DURATION (JOB) MINUTES 00084600 00084700 LET SPT(K)=6 LET SOT(K)=6 00084800 LET XLOC(K) = XCORD(CENTROID(K)) 00084900 LET YLOC(K)=YCORD(CENTRUID(K)) 00085000 REMOVE JOB FROM QUEUE(1) 00085100 RETURN 00085200 LOOP 00085300 FOR I=2 TO N.P.CLASS DO IF QUEUE(I) IS EMPTY GO TO NEXT.Q 00085400 OTHERWISE IF I LE 3 NOW REASSIGN.CAR.TO.JOB(K.F.QUEUE(T)) RETURN 00085500 00085600 HIMUST FIND & SECTOR CAR!! OTHERWISE 00085700 FOR EACH JOB ON QUEUE(I) DO 00085800 LET NHOOD=NBDID(LOCATION(JOB)) 00085900 FOR J=1 TO NUMB.SFCTORS(K) Dn 00086000 IF NHOOD=SECTORS(K,J) 00086100 NOW REASSIGN. CAR. TO. JOR (K. JOB) 00086200 RETURN 00086300 OTHERWISE 00086400 LOOP 00086500 LOOP 00086600 INEXT.Q. LOOP 00086700 PREVENT.PATROL! 00086800 ADD 1 TO N.AVAILABLE 00086900 NOW PLACE . CAR . ON . PATROL (K) 00087000 RETURN 00087100 END

ROUTINE TO REASSIGN.CAR.TO.JOB GIVEN K AND J

This routine is called from the CALL.END event if CAR K is to be dispatched to JOB J, which has been waiting in a QUEUE. It changes the state of the system to reflect this dispatch, calculates the time that the job has spent waiting in queue, and calls the routine ASSIGN to assign the car to the job.

ROUTINE TO REASSIGN. CAR. TO. JOB GIVEN K AND J LET SECT=NBDID(LOCATION(J)) IF PRIORITY(J)=4 LET XLOC(K)=XCORD(LOCATION(J)) LET YLOC(K) = YCORD(LOCATION(J)) REGARDLESS NOW ASSIGN (K.J.) FOR T = 1 TO N.CAR DO LET KK=ADJACENT.CARS(SECT.I) IF KK NE K GO TO LOUPI OTHFRWISE LET SWT(K) =2 IF I GT N.SECTOR.CARS(SECT) LET SWT(K)=3 REGARDLESS GO TO OUT 1L00P11 LOOP OUT LET PRISPRIORITY (J) REMOVE J FROM QUEUE (PRI) LET WAIY.TIME(PRI) = (TIME.V - ENTRY.TIME(J))*1440.0 LET COUNT(PRI)=1 LET SUT(K)=1 RETURN END

ROUTINE TO PLACE.CAR.ON.PATROL GIVEN K

This routine is called from the CALL.END event if there are no jobs in any QUEUE for CAR K to service. First, the car's status is changed to reflect its availability. If the car is already inside its sector, it remains at its current location. (In reality, cars on patrol move around in their sectors more or less at random. Keeping the car where it is -- the random location of the last job it serviced--is meant to imitate this with a minimum of computing effort, since, if the car moves at random, its expected location is its last location.) Otherwise, the car begins to travel back to the centroid of its sector, and a RETURN.TO, SECTOR event is scheduled for the car at a time determined from the distance it must travel and the average travel velocity for Priority 3 jobs. This routine will select blocks for patrol within a car's sector according to the relative frequency of calls for service at each of the blocks in the sector. The patrol frequencies could be modified by inserting into this routine (at line 00089700) a procedure to select the car's location according to a probability distribution associated with the neighborhood or a probability distribution specific to that car. The PREAMBLE would have to be modified to include the appropriate variables, and the desired patrol frequencies would have to be supplied in the initialization deck.

-38-

(A)

| | | 0000400 | |
|----|---|----------|---|
| | ROUTINE TO PLACE.CAR.ON.PATROL GIVEN K | 10069400 | |
| | DEFINE D AS A REAL VARIABLE | 00089500 | |
| | LET SPT(K)=0 LET SOT(K)=0 | 00089600 | • |
| | IF SWT(K)=4 OR SWT(K)=6 LET SWT(K)=0 RETURN ++CAR WAS IN SECTOR | 00089700 | |
| 11 | ALREADY LEAVE HIM THERE !! | 00089800 | |
| | OTHER ALSE | 00089900 | |
| | CREATE & JOB CALLED PATROL RETURN | 00090000 | |
| | LET SWT(K) =1 LET LJ=CENTROID(K) LET LOCATION(PATROL.RETURN)=LJ | 00090100 | |
| | LET ASSIGNMENT (K) = PATROL . RETURN | 00090200 | |
| | LET PRIORITY (PATROL RETURN) = 3 | 00090300 | |
| | LET D = CALCULATE DISTANCE (XCORD (LJ) VCURD (LJ)-XLOC (K) VLOC (K)) | 00090400 | |
| | LET T.START.RESPONSE(K)=TIME.V | 00090500 | |
| | CREATE A PETURNATO SECTOR CALLED R | 00090600 | |
| | IET CARNUMBER (R) =K ITHE CAR NUMBER OF THIS PATROL RETURNET | 00090700 | |
| | SCHEDULE THE RETURN TO SECTOR CALLED R IN DAVELOCITY (3) HOURS | 00090900 | |
| | LET NEWT CUCHT (HI -D | 00090900 | |
| | | 00091000 | |
| | RETURN | 00091100 | |
| | FND | | |

EVENT RETURN. TO. SECTOR GIVEN K

This event, scheduled internally, occurs whenever an available car that is returning from an out-of-sector job arrives at the centroid of its sector. The event changes the status of the system to reflect the car's arrival.

| EVENT RETURN. TO. SECTOR GIVEN K | 00091300 |
|----------------------------------|----------|
| LET J=ASSIGNMEN)(K) | 00091500 |
| FT YLOC(K)=YCORD(CENTROID(K)) | 00091600 |
| LET SWT(K)=0 | 00091700 |
| DESTROY JOB CALLED J | 00091800 |
| RETURN | 00092000 |
| END | 00092100 |

EVENT OUT. OF. SERVICE GIVEN K SAVING THE EVENT NOTICE

This event can either be scheduled internally (meal times will usually be scheduled in this way) or externally (other out-of-service conditions, such as the breakdown of a car, will usually be scheduled as part of the input job stream).

If the event has been scheduled internally, CAR K will be placed out of service for the number of minutes specified by the input variable MEAL. DURATION, and another OUT.OF.SERVICE event will be scheduled for this car in TOUR.LENGTH hours (TOUR.LENGTH is another input variable). If the event has been scheduled externally, the amount of time the car will remain out of service (in minutes) is read from the card in the input job stream that scheduled this event.

-41-If the car scheduled to go out of service is currently available, it is If the car scheduled to go out of service is currently busy, it will EVENT OUT. OF . SERVICE GIVEN K SAVING THE EVENT NOTICE. 00092300 00092400 DEFINE T.OUT. OF. SERVICE AS A REAL VARIABLE 00092500 LET X.MEAL = OUT.OF.SERVICE IF FUNIT. A (OUT. OF. SERVICE) EQ 0 00092600 SCHEDULE AN OUT. OF. SERVICE GIVEN K IN TOUR. LENGTH HOURS 00092700 LET T.OUT.OF.SERVICE=MEAL.OURATION GO AROUND OTHERWISE READ K 00092800 00092900 T.OUT.OF.SERVICE 00093000 + AROUND+ 00093100 DESTROY THE OUT. OF. SERVICE CALLED X. MEAL 00093200 IF SWT(K) GE 2 GO TO DELAY 00093300 **OTHERWISE** 00093400 IF SWT(K)≐l CALL CANCEL.SECTOR.RETURN GIVEN K 00093500 00093600 LET XLOC(K) = XCORD(CENTROID(K)) 00093700 LET YLOC(K) =YCORD(CENTROID(K)) 0093800

immediately placed out of service and the status of the system is changed to reflect this. If returning to its sector, the car's sector return is canceled and it is immediately placed at the centroid of its sector. be placed out of service as soon as it finishes the job. This is accomplished by placing notice of the out-of-service "job" on QUEUE(1). (Priority 1 jobs are never queued, so only out-of-service jobs will appear on QUEUE(1).) If an out-of-service job for CAR K already exists on QUEUE(1), the time of this re w out-of-service job is added to the duration of the previously scheduled job. A flow chart for the OUT.OF. SERVICE event is given in Fig. 7.

REGARDLESS SUBTRACT I FROM N.AVAILABLE LET SWT(K)=6 LET SPT(K)=6 LET SOT(K)=6 CREATE CALL END SCHENULE THIS CALL . END GIVEN K IN T.OUT. OF . SERVICE MINUTES RETURN DELAY FOR EVERY JOB ON QUEUE(1) DO. IF LOCATION (JOH) =K GO TO ALREADY. SCHEDULED OTHERWISE LOOP CREATE JOB LET LOCATION (JOB) =K LFT ENTRY. TTME (JOB) =TIME.V LET OURATION (JOR) =T.OUT. OF. SERVICE LET PRIORITY (JOH) =6 LET STATUS (JOB) =6 FILE JOB IN QUEUE(1) RETURN ALREADY . SCHEDULED . ADD T.OUT. OF . SERVICE TO DURATION (JOB) RETURN END

EVENT FOR END.OF.SIMULATION

This internal event signals the end of a simulation run. It prints several reports that summarize the simulated patrol activity during the run.

00093900

00096000

00094100

00094200 00094300

00094400

00094500

00094600

00094700

00094800 00094900

00095000

00095100

00095200

n0095300

00095400

00095500

00095600

00095700

00095800



-43-

| | | | | 00031900 |
|---------------------------------------|--------------------------|------------|--|-----------|
| AS. REAL | VARTADI | F C | | 00038000 |
| RAY | - VANIANL | C D | | 0003A100 |
| | | | | 00038200 |
| | | | | 00038300 |
| | | | | 00038400 |
| | | | | 00038500 |
| | | | | 00038600 |
| AND INS | TITUTE | | | 00038700 |
| | · · · | | | 00038800 |
| ATION M | ODEL | | | 00038900 |
| | | | | 00039000 |
| AS | Λ 4 | | | 00039100 |
| | | | | 00039200 |
| • | | | | 00039250 |
| A 4 | | | | 00039300 |
| | | | | 00039400 |
| | in the second | | | 00039450 |
| • • • • • • • • • | | | | 00039500 |
| CITAILA | SUMMARY | | | 00039700 |
| | | | 1 | 00039800 |
| | | 1 | | 00030000 |
| NECE | · | | | 00040000 |
| NESP | ONDING | WORK | ING | 00040100 |
| TN | •••• | | | 00040200 |
| 1.4 | 001 | IN | OUT | 00040300 |
| | | | | 00040400 |
| TITZIK | E11 / | | | 00040500 |
| 1.1718.4 | 5))// IME | • V | | 00040600 |
| SECT | NDSIGN A | V | 1997 - 19 | 00040700 |
| •UTTI 17 | 1003(5]+() 111:35/1++ | UI.SFCT. | JOHS (K) | .00040800 |
| •UTTE 77 | - 111 9.31/11 | ME V . | | 00040900 |
| S S S S S S S S S S S S S S S S S S S | 11.101711 | ME. • V • | | 00041000 |
| | | | | 00041100 |

**** * *** 00041200 **** 00041300 FOR J=1 TO 7. DO ADD UTILIZ(K.J)/TIME.V TO X(J) 1 00P 00041400 ADD IN.SECT.JORS(K) TO IJ 00041500 ADD OUT.SECT.JOHS(K) TO OJ 00041600 ADD TIN TO TOTIN 00041700 ADD TOUT TO TOTOUT 00041800 LOOP 00041900 FOR J=1 TO 7 DO LET X(J)=X(J) ZN.CAR LOOP 00042000 LET TOTINATOTIN/N.CAR LET TOTOUT=TOTOUT/N.CAR 00042100 SKIP 1 OUTPUT LINE PRINT 1 DOUBLE LINE WITH IJ,0J,X(1),X(2),X(3),X(4),X(5),X(6), 00042200 00042300 TOTIN+TOTOUT+X(7) THUS 00042400 **** * **** TOTALS #### * *** 00042500 * *** 0.000 * *** 00042600 00042700 FOR I=2 TO N.P. CLASS. DO START NEW PAGE 00042800 FOR J= 1 TO 20; WRITE TITLE(J) AS A 4 00042900 WRITE AS / 00042950 SKIP 2 LINES 00043000 PRINT 1 LINE WITH I AS FOLLOWS 00043100 QUEUEING STATISTICS FOR PRIORITY CLASS ** 00043200 SKIP 2 OUTPUT LINES 00043300 PRINT 2 LINES WITH MQ(I) AND VQ(I) AS FOLLOWS 00043400 AVERAGE QUEUE-SIZE = ***, *** HISTOGRAM OF QUEUE-SIZE 00043500 NO. IN QUEUE FREQUENCY VARIANCE *** 00043600 SKIP | OUTPUT LINE 00043700 FOR J= 1 TO 30 , WITH PROBQ(I, J) GT 0, DO 00043800 PRINT 1 LINE WITH J-1 AND PROBQ(1, J)/TIME.V AS FOLLOWS 00043900 ** 4 444 00044000 LOOP 00044100 00044200 SKIP 2 OUTPUT LINES 00044300 LET T=NWT(I) *MWT(I) 00044400 LET NTOT=NRT(I) 00044500 IF NTOT=0 LET T=0 GO TO PRT 00044600 OTHERWISE LET T=T/NTUT 00044700 PRT: PRINT 3 DOUBLE LINES WITH NWT(I), MWT(I), VWT(I), NTOT, T THUS 00044800 HISTOGRAM OF WAITING TIMES AVERAGE WAITING TIME OF THE *** 00044900 JOBS DELAYED= ***.** 00045000 WAITING TIME FREQUENCY 00045100 VARIANCE 00045200 (MINUTES) AVERAGE WAITING TIME OF ALL **** 00045300 JOBS DISP + D = ***. *** 00045400 FOR J=1 TO 24. WITH HWT(I,J) GT 0. DO 00045500 PRINT 1 LINE WITH 5+J AND HWT(I,J) THUS 00045600 *** C ## 00045700 LOOP 00045800 SKIP 3 OUTPUT LINES 00045900 LET J=1 00046000 PRINT 3 LINES AS FOLLOWS 00046100 CURRENT QUEUE CONTENTS 00046200 POSITION TIME SINCE RECEIPT 00046300 (MINUTES) 00046400 SKIP 1 OUTPUT LINE 00046500 FOR EVERY JOB IN QUEUE(1) . DO 00046600 PRINT 1 LINE WITH J AND (TIME.V-ENTRY.TIME(JOB)) +1440.0 THUS 00046700 888 *** 00046800 ADD 1 TO J 00046900 1 00P 00047000 LOOP 00047100 CALL RESULTS 00047200 END **SIMULATION** 00047300

ROUTINE FOR RESULTS DEFINE MOVOT AS REAL VARIABLES BEGIN REPORT PRINTING FOR I=1 TO N BEGIN HEADING FOR J= 1 TO 20+ WRITE TITLE(J) AS A WRITE AS / SKIP 2 LINES PRINT 1 LINE AS FOLLOWS RESPONSE TIME HISTOGRAMS SKIP 2 OUTPUT LINES END ! HEADING !! PRINT 1 DOUBLE LINE WITH A GROUP OF PRIORITY CLASS ~ ~ ** SKIP I OUTPUT LINE PRINT 2 LINES AS FOLLOWS RESPONSE TIME (MINUTES) FOR J= 1 TO 16, DO PRINT 1 DOUBLE LINE WITH J AND A GRO 4440 <********* **** LOOP SKIP 1 OUTPUT LINE PRINT 1 DOUALE LINE WITH A GROUP OF **** NUMBER OF RESPONSES **** **** PRINT 1 DOUBLE LINE WITH A GROUP OF **.** ** * AVERAGE R.T. **.** PRINT 1 DOUBLE LINE WITH A GROUP OF VARIANCE OF R.T. 444.44 444.44 ***.** END PREPORT START NEW PAGE FOR I= 1 TO 20 , WRITE TITLE(I) AS WRITE AS / SKTP 2 LINES PRINT 1 DOUBLE LINE AS FOLLOWS NE I GHBORHOOD SKIP 2 OUTPUT LINES PRINT 2 DOUBLE LINES AS FOLLOWS PRIORITY--> 5 TC NO. NBD NO. AVG .HT VAR NO. NO. AVG RT VAR VAR SKTP 1 OUTPUT LINE FOR J=1 TO N+NBU DO LET NI=NSRTI(J) LET N2=NSRT2(J) LET LET T=N1+N2+N3+N4 LET MENI#MSRT1 (J) +N2#MSRT2 (J) +N3#MSR LET V=NI+NI+VSRTI(J)+N2+N2+VSRT2(J)+ IF T=0 LET M=0 LET V=0 GO TO PRI LET M=M/T LET V=V/(T#T) PRIM PRINT 1 DOUBLE LINE WITH NBD.NAME MSRT2(J) + VSRT2(J) + N3 + MSRT3(J) + VSRT3 THUS **** ** *** ** ** **** ****

| -45 |
|-----|
|-----|

| | 00047400 |
|---|-----------|
| | 00047500 |
| POLLASS IN GROUPS OF 6 PER PAGE | 00047600 |
| | 00047700 |
| | 00047800 |
| | 00047830 |
| | 00047900 |
| | 00048000 |
| | 00048200 |
| | 00048300 |
| I FIELDS THUS | 00048400 |
| 2. A. K.M. 1945 | 00049500 |
| | 00048500 |
| | 00048800 |
| | 00040700 |
| | 00048800 |
| | 00048900 |
| | 00049100 |
| UP OF FREATLY FIFLAS THUS | 00049200 |
| | 00049300 |
| | 00049400 |
| | 00049500 |
| | 00049600 |
| NRT(I) FIELDS THUS | 00049700 |
| **** | 0049800 |
| | 00049900 |
| MRT(I) FIELDS THUS | 00050000 |
| **** | 00050100 |
| | 00050200 |
| VRT(I) FIELDS THUS | 00050300 |
| *** | 00050400 |
| | 00050500 |
| | 00050600 |
| | 00050700 |
| A A I | 00050800 |
| | 00050850 |
| | 0005090 |
| CHMMARY OF DESDONISES D | V00051000 |
| SUMMART OF HESPONSES B | 00051200 |
| | 00051300 |
| | 00051400 |
| 2 3 | 00051500 |
| DTALS | 00051600 |
| AVG RT VAR NO. AVG RT | 00051700 |
| AVG RT VAR | 00051800 |
| | 00051900 |
| | 00052000 |
| N3=NSRT3(J) LFT N4=NSRT5(J) | 00052100 |
| | 00052200 |
| (T3(J)+N4*MSRT5(J) | 00052300 |
| N34N34V5RT3(J)+N4#N4+VSRT5(J) | 00052400 |
| OTHERWISE | 00052500 |
| | 00052600 |
| | 00052700 |
| (J) + N1 + MSRT1(J) + VSRT1(J) + N2 + | 00052800 |
| $(J) \bullet N \bullet \bullet MSRT5 (J) \bullet VSRT5 (J) \bullet T \bullet M \bullet V$ | 00052900 |
| | 000000000 |
| · · · · · · · · · · · · · · · · · · · | 000E3300 |
| ****** | 00033600 |

1.00P 00053300 SKIP I OUTPUT LINE 00053400 LET T=0 LET M=0 LET V=0 00053500 FOR 1=1 TO 4 DO 00053600 IF I EQ 4. LET I = 5 REGARDLESS 00053700 LET N=NRT(I) ADD N TO T 00053800 ADD N#MRT(I) TO M ADD N#N#VRT(1) TO V 00053900 LOOP 00054000 IET MEM/T LET V=V/(T+T) 00054100 PRINT 1 DOUBLE LINE WITH NRT(1) + MRT(1) + VRT(1) + NRT(2) + MRT(2) + 00054200 VRT (2) +NRT (3) +MRT (3) +VRT (3) +NRT (5) +MRT (5) +VRT (5) +T+M+V THUS 00054300 **** TOTALS *** *** +00054400 **** ****** ***** *.*** 00054500 SKIP 9 OUTPUT LINES 00054600 PRINT 2 LINES AS FOLLOWS 00054700 PICK-UP (PRIORITY 4) 00054800 NO. AVG RT VAR 00054900 NBD SKIP 1 OUTPUT LINE 00055000 FOR J=1 TO N.NBD DO 00055100 LET N5 = NSRT4(J) 00055200 PRINT 1 LINE WITH NBD.NAME (J) +N5+MSRT4 (J) +VSRT4 (J) AS FOLLOWS 00055300 **** *** 00055400 1.00P 00055500 SKIP I OUTPUT LINE 00055600 PRINT 1 LINE WITH NRT(4), MRT(4), VRT(4) THUS 00055700 **** ***** **** TOTALS 00055800 START NEW PAGE 00055900 FOR I= 1 TO 20 . WRITE TITLE(I) AS 00056000 A 4 WRITE AS / 00056050 SKIP 2 LINES 00056100 PRINT 1 LINE AS FOLLOWS 00056200 DISTRIBUTION OF CAR AVAILABILITY 00056300 SKIP 2 OUTPUT LINES 00056000 PRINT 2 LINES AS FOLLOWS 00056500 NO. OF CARS PER CENT 00056600 ON PATROL OF TIME 00056700 FOR TEO TO NOCAR DO 00056800 PRINT 1 LINE WITH I AND 100*HISTAV(I+1)/TIME.V THUS 00056900 ** 4.4** 00057000 1.00P 00057100 SKIP 1 OUTPUT LINE 00057200 PRINT 1 LINE WITH MAV THUS 00057300 AVERAGE NUMBER AVAILABLE = **. *** 00057400 PRINT I LINE WITH VAV THUS 00057500 VARIANCE =+++. +++ 00057600 START NEW PAGE 00057700 FOR I= 1 TO 20 . WRITE TITLE(I) AS Δ 4 00057800 WRITE AS / 00057850 SKIP 2 LINES 00057900 PRINT | LINE AS FOLLOWS 00058000 NUMBER OF PRECINCT CARS SENT TO CALLS 00058100 SKIP 2 OUTPUT LINES 00058200 PRINT 1 DOUALE LINE THUS 00058300 PRTORITY 2 00058400 0 1 З 10 11 12 13 14 15 AVERAGE VARIANCE 00058500 WRITE AS / ... 11 00058600 1 FOR I=1 TO 16.WRITE HISTC(1.1) AS I 6 00058700 WRITE MC(1) AND VC(1) AS S 6,0(6,2),5 5,0(7,2),/ 00058800 FOR 1=2 TO N.P.CLASS. DO 00058900 PRINT DOUBLE LINE WITH I HISTC(I . 1) HISTC(I,2) HISTC(I,3) . 00059000 HISTC(I+4)+MC(I)+ AND VC(I) THUS 00059100 **** 00059200 00059300 黄黄菊 。华华 *** LOOP 00059400 STOP 00059500 END 00059600 00059700

-46-

There are two basic sets of data needed to run the simulation. The first input data set, called the "initialization deck," follows the job control cards and specifies the geographical configuration of the region being simulated, the patrol unit assignments, and initial parameter values such as the number of hours in a tour, response velocities, meal times, the duration of the simulation, etc. (The initialization deck is the GO.SYSIN data set.) The second data set is the series of calls for service to be responded to during the course of the simulation, and is referred to as the "job stream." (It is the SIMU04 data set.) In what follows, we describe, card by card, all the data needed for the simulation. For easy reference to the program, the variable name used to identify each piece of data in the simulation is given in upper case in parenthesis. So, for example, the simulation variable N.CAR is used to refer to the number of patrol units assigned to the region being simulated.

The Initialization Deck (GO.SYSIN)

- (5.1) Card 1--supplies the title for the simulation run. This title card).
- (5.2) Card 2--a description of the region being simulated (e.g., 26th ters long (the contents of one complete card).
 - - Priority 1 incident (MAX.SENT).
 - (MEAL. DURATION) in minutes.

V. THE INPUT DATA

-47-

will be printed at the top of every output report. The title can be up to 80 characters long (the contents of one complete

Precinct, 7th Division, etc.). This is printed on the first page of the output report. The description can be up to 80 charac-

(5.3) Card 3--supplies the initial values for four system parameters: (a) The length of the simulation (SIM.LENGTH) in hours.

(b) The maximum number of cars that will be dispatched to a

(c) The duration of an internally scheduled OUT.OF.SERVICE job

(d) The length of a single tour of duty (TOUR.LENGTH) in hours. This variable is used to schedule an OUT.OF.SERVICE event for each car one time during every tour of duty (e.g., for a meal). After the first internally scheduled OUT.OF. SERVICE event for a car (scheduled at a time specified as

described in paragraph (5.7) below), an OUT.OF.SERVICE event for the car will occur every TOUR.LENGTH hours.

(5.4) Card 4--the number of BLOCKs (N.BLOCK) in the region being simulated (A BLOCK is the smallest geographical unit in the simulation), the number of neighborhoods (N.NBD), and the number of patrol units (N.CAR). These three integers are punched in three fields, separated by at least one blank space. Note that, for internal reference, the blocks will be numbered 1 to N.BLOCK, the neighborhoods from 1 to N.NBD, and the cars from 1 to N.CAR.

-48-

(5.5) N.BLOCK Cards--one for each block, containing:

K

- (a) The block number used as an internal reference number for the block (B). These must run in sequence from 1 through N. BLOCK.
- (b) The external identification number associated with the block (TAXNO(B)). This number can be its tax block number, census block, or any other identifying number.
- (c) The internal reference number of the neighborhood to which the block belongs (NBDID(B)).
- (d) The x and y coordinates of the center of the block (XCORD(B), YCORD(B)).

This information is punched in five fields, each separated by at least one blank space. The variables must be integers, except for the coordinates, which can be punched with decimal points. We emphasize that NBDID(B) is a number and not an alphanumeric sector name associated with the neighborhood. Further, NBDID(B). must correspond to the appropriate internal neighborhood sequence number (N) used when neighborhood information is being input. (See paragraph (5.6) below.) These N.BLOCKs may comprise a separate data set. If so, they must be kept separate from the GO.SYSIN data, and a data definition (DD) card for the additional data set must be included in the job control cards. In addition, the statement in the MAIN routine that reads this data (statement number 34200) should indicate the logical unit from which this data should be read (e.g., the version given here reads this data from logical unit ?).

(5.6) N.NBD Sets of Cards -- one set for each neighborhood, consisting

of:

- by blanks:

 - respond to the internal car numbers (C), discussed in
- paragraph (5.7) below.
- of:
- by blanks:

 - the unit (CAR.NAME(C)).

-49-

(a) One card containing the following four fields, separated

(1) The internal neighborhood reference number (N). These must run in sequence from 1 to N.NBD. (2) An alphanumeric sector name associated with the neighborhood (NBD.NAME(N)).

(3) The number of sector cars assigned to the neighborhood (N.SECTOR.CARS(N)).

(4) The number of patrol units designated as adjacent resources for the neighborhood (N.ADJACENT(N)). (b) One or more cards containing a list of all the cars in the simulation in the order in which they will be nominated for dispatching in this neighborhood (ADJACENT.CARS(N.J). The members of the list are car numbers, punched in N.CAR fields, each separated by at least one blank space. The car numbers used must cor-

(5.7) N.CAR Sets of Cards--one set for each patrol unit, consisting

(a) One card containing the following five fields, separated

(1) The internal reference number for the unit (C). These must run in sequence from 1 to N.CAR. (2) The 4-character alphanumeric name associated with

(3) The number of neighborhoods to which the unit is assigned as a "sector" car (NUMB.SECTORS(C)). If none (e.g., a supervisor's car), put a 1 in this field and a zero in the following card.

(4) The internal block number of the block at which the unit is "stationed" while on patrol (CENTROID(C)). This number must correspond to one of the block numbers (B) discussed in paragraph (5.5) above.

(5) The scheduled time into a tour (in decimal hours, e.g., 1.5) at which the car will be placed out of service (MEAL.TIME).

- (b) One or more cards containing the list of neighborhoods to which the unit is assigned as a "sector car." This list should contain NUMB.SECTORS(C) entries, which are the internal neighborhood reference numbers (N) desscribed in paragraph (5.6) above. Overlapping sectors are achieved by assigning some or all of the same neighborhoods to two or more cars.
- (5.8) 1 Card--giving the response velocity in miles per hour for each of the five priority classes (VELOCITY(P)), punched as real numbers in five fields, each separated by at least one blank space, in order of priority class, starting with Priority class 1.
- (5.9) 1 Card--giving the proportion of the job duration (P.DURATION (I)), assigned to each of the three types of responding units (primary, backup, and tertiary), punched as real numbers in three fields, each separated by at least one blank space.

The Job Stream (SIMU04)

Ê

- (5.10) In chronological order, as many cards as there are calls for service and externally scheduled OUT.OF.SERVICE events. For each call for service, the card must contain the following 9 fields describing the job:
 - (1) The event name JOB.ENTRY punched in columns 1-9.
 - (2-4) The time the job enters the system (ENTRY.TIME(J)), measured in days, hours, and minutes from the start of the simulation, punched in three fields.
 - (5) The location of the job (LOCATION(J)), given by the internal block number. This number must correspond to one of the values of B given in paragraph (5.5) above.
 - (6) The priority of the job (PRIORITY(J)).
 - (7-8) The duration of the job in hours and minutes, punched in two fields (HDUR and MDUR).
 - (9) An asterisk (*).

All fields must be separated by at least one blank space. There should always be at least one job whose entry time is later than SIM.LENGTH, the end of the simulation.

For each OUT.OF.SERVICE event, the card must contain the following three fields of information: (1) The event name OUT.OF.SERVICE punched in columns 1-14.

AN EXAMPLE

We illustrate the data input with sample data sets for a simulation of the 71st Precinct in the New York City Police Department. The 71st Precinct has been broken down into 305 blocks, which, in this simulation run, are grouped into 12 neighborhoods. Each neighborhood corresponds to one of 12 nonoverlapping sectors. Figure 8 is a street map of the 71st Precinct and Fig. 9 is a computer-generated map that uses the coordinate system of the simulation and shows each individual tax block and the sector to which it belongs. In this example, there are seven patrol units: six are sector cars, each of which is assigned to a pair of neighborhoods, and one is a sergeant's car. Primary cars work for the entire duration of the incident; backup cars work for half the duration of the incident, and tertiary cars are released as soon as they arrive at the incident. Responses to all incidents, no matter what their priority, are made at a velocity of 20 mph. The data set describing each of the blocks is read from logical unit 7. A listing of the control cards and initialization deck for this example is given in Appendix B. Some of the key data elements are explained in Tables 5 and 6. Appendix C lists the first few cards in a sample job stream for a 16-hour simulation of this region.

(2) The internal indentification number of the car to be placed out of service. This number must correspond to one of the values of C given in paragraph (5.7) above. (3) The length of the out-of-service time in minutes.





VI. OUTPUT REPORTS

The simulation program produces several types of output. We illustrate the various output reports by showing those produced by a 16-hour simulation of New York City's 71st Precinct, using the sample input data presented in Appendix B. First, after reading the initalization deck, the program prints out the title and the following input data;

- The length of the simulation (in hours).
- being simulated.
- cars constituting its set of ADJACENT.CARS.

A sample of this initialization output is shown in Figs. 10 and 11. After simulating the patrol operations in the region for the required number of hours, the program prints a series of six reports that summarize the simulated activity. Each of these reports is described below.

(1) CAR ACTIVITY SUMMARY (Fig. 12)

A summary is given for each car, showing how it spent its time during the course of the simulation. The following information is printed for each car:

- The number of jobs to which it responded within its sector and the number to which it responded outside its sector.
- outside job).

| | Table | 5 | |
|--------|-------------|-----|------------|
| SAMPLE | SIMULATION: | CAR | PARAMETERS |

-54-

| Car Number (C) | Car Name (CAR.NAME(C)) | Number of Sectors (NUMB, SECTORS (C)) | Meal Time (MEAL.TIME) | Patrol Block CENTROID(C)) | Assigned SECTORS(C,J)) |
|----------------------|---|---|-----------------------------|---------------------------------|---------------------------|
| 1 | AB | 2 | 3 | 144 | 1,2 |
| 2 | CD | 2 | 4 | 222 | 3,4 |
| 3 | EF | 2 | 5 | 100 | 5,6 |
| 4 | HI | 2 | 3 | 43 | 8,9 |
| 5 | JG | 2 | 4 | 296 | 7,10 |
| 6 | KM | 2 | 5 | 63 | 11,12 |
| 7 | SGT | 0 110 | 4 | 59 | 0 |
| | Anna para interna de la companya de | <u> </u> | . . | | L |

Table 6 SAMPLE SIMULATION: NEIGHBORHOOD DATA

| Neighborhood | Neighborhood | NeighborhoodNumber of Sector CarsNameSector Cars(NBD.NAME(N))N.SECTOR.CARS(N)) | | Adjacent | | | |
|---|--|---|--|---|--|--|--|
| Number | Name | | | Car List | | | |
| (N) | (NBD.NAME(N)) | | | (ADJACENT,CARS(N,J)) | | | |
| 1 2 3 4 5 6 7 8 9 10 11 12 | A B C D E F G H I J K M | 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | | (1, 2, 3, 4, 5, 6, 7) $(1, 2, 3, 4, 5, 6, 7)$ $(2, 3, 4, 5, 6, 1, 7)$ $(2, 3, 4, 5, 6, 1, 7)$ $(3, 1, 4, 2, 6, 5, 7)$ $(3, 1, 4, 2, 6, 5, 7)$ $(3, 1, 4, 2, 6, 5, 7)$ $(5, 2, 6, 4, 3, 1, 7)$ $(4, 6, 3, 2, 5, 1, 7)$ $(4, 6, 3, 2, 5, 1, 7)$ $(5, 2, 6, 4, 3, 1, 7)$ $(6, 5, 4, 2, 3, 1, 7)$ | | | |

-55-

• The number of blocks, neighborhoods, and patrol cars in the region

• A listing of the data associated with each block, including its internal reference number, external identification number, x and y coordinates; and the data associated with each neighborhood, including its internal reference number, the number of sector cars and adjacent cars assigned to it, and the ordered list of patrol

• The proportion of time the car spent on preventive patrol in its sector and the proportion of time it was available to respond from outside its sector (while returning to its sector from an

 The proportions of its time spent responding to calls that occurred inside its sector and responding to calls outside its sector.

- The proportions of its time spent working on jobs in its sector and working outside its sector.
- The proportion of its total in-service time spent within its sector and the proportion of time spent out of its sector.
- The proportion of the total simulated time during which the car was out of service.

The last line of the report prints the values of each of the above quantities, averaged over the N.CAR patrol units in the simulation.

(2) QUEUEING STATISTICS (Fig. 13)

A one-page report is printed for each priority class having a queue (i.e., all classes except Priority 1). This report contains three types of information:

- Queue sizes -- a frequency histogram, plus the average and variance of the queue size. (The histogram shows frequencies for queues of up to 50 jobs.)
- Waiting times -- a frequency histogram of the waiting time of jobs that were queued before being dispatched, the average and variance of the waiting time for these jobs, and the average waiting time for all jobs dispatched, including those that were not delayed. (The histogram shows waiting time frequencies in five-minute intervals up to four hours.)
- Current (final) queue contents -- a list of those calls in queue and waiting to be dispatched at the instant the simulation ends, together with the time elapsed since each call was received (the time each call has already spent in queue).

(3) RESPONSE TIME HISTOGRAMS (Fig. 14)

A report listing, for each priority class, the number of calls that had response times falling into each one-minute interval (i.e., 0-1 minute, 1-2 minutes, etc.) up to sixteen minutes, where response time is defined as the time from dispatch of call until the arrival of the first patrol car. In addition, the total number of responses is printed for each priority class.

(4) NEIGHBORHOOD ACTIVITY SUMMARY (Fig. 15)

This report shows the workload (number of jobs) and response time information for each neighborhood, broken down by priority class. Thus, the number of jobs responded to, together with the average and variance of the response times to these jobs, is printed for each priority class, neighborhood by neighborhood. Totals are printed for each neighborhood and each priority class.

(5) DISTRIBUTION OF CAR AVAILABILITY (Fig. 16)

This report shows the percentage of time that a given number of cars were on preventive patrol in the region, from O patrol units to N.CAR patrol units. In addition, the average number of available cars and the variance are printed.

(6) NUMBER OF PRECINCT CARS SENT TO CALLS (Fig. 17)

Since under different dispatching rules different numbers of cars may be sent to calls, this report shows how often any given number of cars (from 0 to 15) are sent to calls in each priority class. In some priority classes exactly one car is always sent (e.g., Classes 3, 4, and 5), but Priority 1 calls receive up to MAX.SENT of the available cars. The number of Priority 1 calls that received zero cars indicates the number of times a Priority 1 call was received when all N.CAR patrol units in the region being simulated were busy.

-56-

-57-

6 PATROL CARS AND 1 SERGEANT'S CAR

| SIMULATION LENGTH = 16.00 HOURS | | NRD ID | SECTORS | NUMBER OF | ORDER IN WHICH |
|------------------------------------|--|--|--|--|--|
| NO. OF BLOCKS = 305 NO. OF NBDS. = | 12 NO. OF CARS = 7 | | an a | SECT. CARS ADJ. CARS | CARS ARE NOMINATED |
| | n an | | A | 1 1 | |
| 1 1185 12 | 1170 2.18140 | | | | 2 |
| | 9450 2.12950 | | | | 3 |
| 3 1188 12 .76 | 6100 2-07780 | | | | $\mathbf{\tilde{4}}$ |
| 4 1189 12 .76 | 5830 2.00850 | | | | $\mathbf{S}_{\mathbf{s}}$ |
| 5 1190 12 .75 | 5940 1.94630 | | | | Ğ |
| 6 1192 12 .74 | 4620 1.83550 | | | | 7 |
| 7 1196 12 .73 | 3350 1.73110 | 2 | 8 | and the b oy of the state of t | |
| 8 1197 11 .69 | 9360 1.65230 | | | | 1 |
| 9 1198 11 .71 | 1180 1.58130 | | | | 2 |
| 10 1266 12 .96 | 5310 2.12540 | | | | 3 |
| 11 1267 12 1.08 | 3660 2.09430 | | | | 4 |
| 12 1268 9 1.21 | 1150 2.07400 | | | | 5 |
| 13 1269 9 1.35 | 5220 2.05450 | | | | 6 |
| 14 1270 9 1.49 | 9190 2.03140 | 3 | С | 1 | 1 |
| 15 1271 9 1.63 | 3190 2.01260 | | | | 9 |
| 16 1272 6 1.77 | 7100 1.98720 | | | | 2 |
| 17 1273 12 .94 | 4520 2.07420 | | | | 4 |
| 18 1274 12 1.07 | 7370 2.04180 | | | | 5 |
| 19 1275 9 1.20 | 0350 2.02170 | | | | 6 |
| 20 1276 9 1.34 | 4540 2.00060 | | | | 1 |
| 21 1277 9 1.48 | 8500 1.97950 | | a af a sea | | 7 |
| 22 1278 9 1.62 | 2440 1.95770 | 4 | , D.A | 1 | |
| 23 12/9 0 1.76 | 5420 I.JJD00 | | | | 2 |
| | 2480 2.01610 | | | | 3 |
| | | | | | 4 |
| | 75411 1.902411 | | | | 5 |
| | 3520 1.94100 7500 1.59010 | | | | |
| | (580) 1.92010 1570 1.90770 | | | | |
| | 10/11 1007/10 | 5 | E | 1 1 | |
| | 040U 100773U | | | | a |
| | | | | | $\mathbf{v}_{1} = \mathbf{v}_{1} + \mathbf{v}_{2} + \mathbf{v}_{1} + \mathbf{v}_{2} + \mathbf{v}_{3} + \mathbf{v}_{4} $ |
| | 8560 1.90230 | | | | $\overline{4}$ |
| 34 1290 8 1.32 | 2660 1.88150 | | | | 2 |
| 35 1291 8 1.46 | 5690 1.86140 | | | | 6 |
| 36 1292 8 1.60 | n610 1.83970 | | | | 5 |
| 37 1293 5 1.74 | 4740 1.81760 | ана (1997) Спорта страна (1997) Спорта страна (1997) | - | | 7 |
| 38 1294 12 | 3370 1.90050 | ана на селото на селото на Б и | - F | $\mathbf{I}_{\mathbf{I}}$, $\mathbf{I}_{\mathbf{I}}$ | |
| 39 1295 12 1.02 | 2960 1.86610 | | • | | 3 |
| 40 129601 8 1.14 | 4210 1.84800 | | | | |
| 41 129602 8 1.21 | 1210 1.83870 | | | | 4 |
| 42 1297 8 1.31 | 1790 1.82290 | | | | С. С |
| 43 1298 8 1.45 | 5730 1.79970 | | | | р 5 |
| 44 1299 8 1.59 | 9730 1.78100 | | | | 7 |
| 45 1300 5 1.73 | 3730 1,75870 | 7 | G | 1 | |
| 46 1301 12 ,80 | 0640 1.82220 | | | | 5 |
| | | | | | |

-58-

Input data for the 71st Precinct, Brooklyn, New York

Input data for the 71st Precinct, Brooklyn, New York

THE NEW YORK CITY-RAND INSTITUTE POLICE PATROL SIMULATION MODEL

-60-

Ë 1

SIMULATION RESULTS FOR THE 71ST PRECINCT, BROOKLYN, NEW YORK

6 PATROL CARS AND 1 SERGEANT'S CAR

- 37, 58

| CAR | NO. OF | JOBS | ON | PATROL | RESI | PONDING | WORK | ING | TOT. IN | SERVICE | | | |
|--------|--------|------|------|--------|------|---------|------|------|---------|---------|-----|------------|---|
| | IN | OUT | IN | OUT | IN | OUT | IN | OUT | IN | OUT | OUT | OF SERVICE | |
| AB | 8 | 6 | .311 | .009 | .014 | .031 | .207 | .196 | •533 | .236 | | .231 | |
| CD | 5 | 16 | .297 | .017 | .009 | • 054 | .120 | .339 | .426 | .410 | | .164 | |
| EF | 8 | 11 | .272 | .016 | .013 | .032 | •512 | .255 | .499 | .303 | | .198 | |
| HI | 5 | 16 | .232 | .023 | .009 | .055 | .093 | .428 | .333 | .506 | | .160 | ÷ |
| JG | 8 | 12 | .362 | .026 | .015 | .039 | .210 | .183 | •58B | .248 | | .165 | |
| KM | 8 | 10 | •296 | .017 | .012 | •038 | •257 | •535 | •565 | .287 | | .148 | |
| SGT | 0 | 12 | .465 | .016 | 0. | •032 | 0• | .321 | .465 | .369 | | .167 | |
| TOTALS | 42 | 83 | .319 | .018 | .010 | •040 | •157 | .279 | .487 | .337 | | .176 | |
| | | | | | | | | | | | | | |

CAR ACTIVITY SUMMARY

Fig. 12-Car activity summary

6 PATROL CARS AND 1 SERGEANT'S CAR

QUEUEING STATISTICS FOR PRIORITY CLASS 3

HISTOGRAM OF QUEUE-SIZE NO. IN QUEUE FREQUENCY

012345

6 7, 8 AVERAGE QUEUE-SIZE = VARIANCE .691 2.314

.758 .092 .031 .029 .030 .036

•000 •000

| H | IST | OGRAM | OF | WAITING | 5 TIMES |
|-----|----------------|-------|----|---------|----------|
| W۵ | ITI | NG TI | ME | FF | REQUENCY |
| . (| MIN | UTESI | | | |
| | < | 5 | | | 8 |
| | < ¹ | 10 | | | 9. |
| | < | 15 | | | 9 |
| | · < | 20 | | | 5 |
| | · < | 25 | | | 3 |
| | < | 30 | | | 5 |
| | . < | 35 | | | 3 |
| | · < | 40 | | | 2 |
| | < | 45 | | | 1 . |
| | | | | | |

| AVERAGE | WAITING | TIME | 0F | THE | 45 | JOBS | DELAYED |)≖: | 15.877 |
|---------|---------|------|----|-----|-----|------|---------|-----|---------|
| | | | | | | VARI | ANCE | = | 123.109 |
| AVERAGE | WAITING | TIME | OF | ALL | 104 | JOBS | DISPID | = | 6.870 |

CUPRENT QUEUE CONTENTS SITION TIME SINCE RECEIPT (MINUTES) POSITION

Fig. 13-Queuing statistics for Priority Class 3

6 PATROL CARS AND 1 SERGEANT'S CAR

RESPONSE TIME HISTOGRAMS

| PRIORITY CLASS | 1 | 2 | 3 | 4 | 5 |
|---------------------|-------|-----|----------|-----|-----|
| RESPONSE TIME | | | | | |
| (MINUTES) | | | | | |
| < 1.00 | 1 | 0 | 4 | · 0 | 0 |
| < 2.00 | 2 | 0 | 29 | 0 | 0 |
| < 3.00 | 2 | 0 | 26 | 0 | 0 |
| < 4.00 | 3 | 0 | 17 | Ō | 0 |
| < 5.00 | 1 | 0 | 11 | 0 | Ő |
| < 6.00 | ō | 0 | Q | õ | Õ |
| < 7.00 | Ő | n | 5 | ň | ů ů |
| < 8.00 | Ô | Ň | 2 | Ö | 0 |
| < 9.00 | 0 | n n | 2 | 0 | 0 |
| <10.00 | 0 | 0 | 1 | · | 0 |
| | 0 | 0 | | 0 | 0 |
| <12 00 | 0 | 0 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 0 | 0 |
| | | 0 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 0 | 0 |
| <15.00 | 0 0 U | 0 | 0 | 0 | 0 |
| <16.00 | 0 · | 0 | 0 | 0 | 0 |
| NUMBER OF RESPONSES | Q | n | 104 | 0 | 0 |
| AVERAGE P.T. | 2.11 | 0 | 2.62 | • | • |
| VARIANCE OF R.T. | 1.43 | 0. | 3.00 | 0. | 0. |

-62-

Fig. 14-Response time histograms

6 PATROL CARS AND 1 SERGEANTIS CAR

SUMMARY OF RESPONSES BY NEIGHBORHOOD

| | POTOS | | · | | | 2 | | | | | | | | | TOTALE | |
|-----|-------|---------|-------|------|-------|--------|-----|-----|--------|-------|-------|-------|-----------|-----|--------|-------|
| NBD | | NO. | AVGRT | VAR | NO. | AVGERT | VAR | NO. | AVGRT | VAR | NO. | AVGRT | VAR | NO. | AVG RT | VAR |
| Δ. | | 0 | 0. | 0. | 0 | 0. | 0. | 10 | 4.000 | 3.000 | Ċ | 0. | 0. | 10 | 4.000 | 3.000 |
| B | | 2 | 2.000 | 0. | 0 | 0. | 0. | 16 | 2.938 | 4.559 | 0 | 0. | 0. | 18 | 2.833 | 3.602 |
| С | | 0 | 0. | 0. | 0 | 0 | 0. | 7 | 2.286 | 1.347 | · · C | 0. | 0. | 7 | 2.286 | 1.347 |
| D | | 0 | 0. | 0. | 0 | 0 | 0. | 6 | 2.167 | 2.139 | Ċ | 0. | 0. | 6 | 2.167 | 2.139 |
| E | | 1 | 0. | 0. | 0 | 0. | 0. | 7 | 2.000 | 2.857 | · · 0 | 0. | 0. | 8 | 1.750 | 2.187 |
| F S | | Ó | 0. | 0. | Ō | 0. | 0. | 6 | 2.333 | 2.889 | C | 0. | 0. | 6 | 2.333 | 2.889 |
| G | | 1 | 3.000 | 0. | 0 | 0. | 0. | 8 | 2.500 | 2.000 | C | 0. | 0. | 9 | 2.556 | 1.580 |
| H | | 5 | 3.500 | .250 | 0 | 0. | 0. | 3 | 1.667 | 2.889 | 0 | 0. | 0. | 5 | 2.400 | 1.080 |
| T - | | · · · 0 | 0. | 0. | 0 | 0. | .0. | 8 | 2.125 | .859 | 0 | 0. | 0. | 8 | 2.125 | . 959 |
| J | | 1 | 1.000 | 0. | 0 | 0. | 0. | 15 | 3.400 | 3.973 | 0 | 0. | 0. | 16 | 3.250 | 3.492 |
| K. | | 1 | 1.000 | 0. | 0 | 0 | 0 | 12 | 2.083 | .743 | 0 | 0. | 0. | 13 | 2.000 | .633 |
| 14 | | 1 | 3.000 | 0. | . 0 | 0. | 0. | 6 | 1.833 | .472 | C | 0. | 0. | 7. | 5.000 | .347 |
| | | | 22000 | ••• | · · · | ~ - | | | 1.0000 | | | | UU | | | . ' |



6 PATROL CARS AND 1 SERGEANT'S CAR DISTRIBUTION OF CAR AVAILABILITY

| NO. OF CARS | PER CENT |
|-------------|----------|
| ON PATROL | OF TIME |
| 0 | 29.886 |
| 1 | 8.427 |
| 5 | 14.250 |
| 3 | 13.111 |
| 4 | 18.421 |
| 5 | 10.210 |
| 6 | 4.974 |
| 7 | .721 |
| | |

AVERAGE NUMBER AVAILABLE = 2.359 VARIANCE = 3.914

Fig. 16–Distribution of car availability

Fig. 17-Number of precinct cars sent to calls

| | VARIANCE | 6 7 7 0 0 0 0 0 |
|-----------|----------|--------------------------------------|
| | AVERAGE | 2.10 2.00 0.00 0.00 |
| | 15 | 0 |
| | 14 | 0 |
| | 13 | 0 |
| | 12 | 0 |
| | 11 | 0 |
| | 10 | 0 |
| | 0 | C |
| | œ | C |
| | ~ | 0 |
| | 9 | 0 |
| | ស | C |
| ALLS | 4 | 0 |
| T TO C | e | 00000 |
| S SEN | N | 60000 |
| VCT CAI | - | MOGOO O I |
| PRECIN | 0 | |
| NUMPER OF | PRIORITY | |

S CAR

SERGEANT

AND 1

6 PATROL CARS

-64-

F

0

Č

n,

Appendix A

0

.

Ő.

Note that the program begins with a section called the PREAMBLE. This section plays a special role in any SIMSCRIPT II program. In it, one defines variables, entities, attributes, lists, and events, specifies the number of dimensions for arrays, and makes provision for gathering statistics. Lines 25000 through 31700 constitute the PREAMBLE of this simulation. The PREAMBLE is followed by the MAIN routine (lines 31900-37800), which is followed by the events and subroutines that comprise the simulation program.

-67-

14

LISTING OF THE SIMULATION PROGRAM

| PREAMBLE | 0,0025000 |
|--|-----------------------|
| LAST COLUMN IS 72 | 00025100 |
| EXTERNAL EVENT UNIT IS 4 | 00025200 |
| NORMALLY MODE IS INTEGER AND DIMENSION IS 0 | 00025300 |
| DEFINE \$ TO MEAN IN ARRAY | 00025400 |
| | 00025500 |
| PERMANENT ENTITIES | 00025600 |
| EVERY BLOCK HAS A TAXNO 5 1. AN XCORD 5 2. A VOORD 5 3 AND A NBDID 5 | 400025700 |
| EVERY NOD HAS A NOD-NAME \$ 5, AN N. ADJACENT \$ 6, | 00025800 |
| AN N.SHURDR.CARS \$ 7. A PI.NBD.RSPONSE \$ 24. A P2.NBD.RSPONSE \$ 257 | 00025900 |
| A P3.NBU.RSPONSE \$ 26, A P4.NBD.RSPONSE \$ 27, | 00026000 |
| AND A PS.NBD.RSPONSE \$ 28 | 00026100 |
| EVERY CAR HAS A CAR.NAME \$ 8, AN XLOC \$ 9, A YLOC \$ 10, A PRI.CAR \$ 11 | 00026200 |
| A NUMB-SECTORS & 12+AN ASSIGNMENT \$ 13+A T_START-RESPONSE \$ 14+ | 00026300 |
| A NEXT.EVENT \$ 15, A (SPT(1/4),SOT(2/4),SWT(2/2)) \$ 16, | 00026400 |
| A CENTROID & 17+AN IN.SECT.JOBS & 18,AN OUT.SECT.JOBS \$ 19. | 00026500 |
| AND MAY BELONG TO AN ASSIGNED.CARS | 00056600 |
| EVERY P.CLASS OWNS A QUEUE, AND HAS A JOB.RESPONSE.TIME \$ 20. | 00026700 |
| A VELOCITY \$ 21.A COUNT \$ 22. AND A WAIT.TIME \$ 23 | 00026800 |
| | 00026900 |
| DEFINE XCORD, YCORD, XLOC, YLOC, TOSTART, RESPONSE, VELOCITY, WAIT. TIME | 00027000 |
| AS REAL VARIABLES | 00027100 |
| DEFINE NBD.NAME AND CAR.NAME AS ALPHA VARIABLES | 00027200 |
| DEFINE ADJACENT.CARS AND SECTORS AS 2-DIMENSIONAL ARRAYS | 00027300 |
| DEFINE P.DURATION AS A 1-DIMENSIONAL REAL ARRAY | 00027400 |
| DEFINE OUTSIDE.DISPATCH AS AN INTEGER VARIABLE | 00027500 |
| DEFINE N.AVAILABLE AS AN INTEGER VARIABLE | 00927600 |
| DEFINE MAX.SENT, MEAL.DURATION, AND TOUR, LENGTH AS INTEGER VARIABLE | 500027700 |
| DEFINE REGION NAME AND TITLE AS I-DIMENSIONAL ALPHA APPAYS | 00027300 |
| | 00027900 |
| ILMPURANT ENTITIES | 00028000 |
| A DURATION HAS A LUCATIONAAN ENTREATMERA PRIORITEA STATUSA | 00020100 |
| A DURATION (MAT BELUNG TO A BUEVE SAND UWNS AN ASSIGNED CARS | 60020200 |
| DEFINE ASSIDNED CARS AS A FIFU SET WITHOUT FORFARR AND RE RUUTINE | 300023300 30022400 |
| SETINE AVENUE AND DURATION AS REAL VARIABLES | 00028500 |
| DELINE FRAMMALINE HAD DOWNLING HO WERE ARVINDED | 00028600 |
| EVENT NOTICES INCLUDE END. OF. SIMULATION | 00028700 |
| EVERY ARRIVAL AT SCENE HAS A CARNUMBER IN WORD 6 | 00028800 |
| EVERY CALL END HAS A CARNUMHER IN WORD 6 | 00028900 |
| EVERY OUT OF SERVICE HAS A CARNUMBER IN WORD 6 | 00029000 |
| EVERY RETURN. TO. SECTOR HAS A CARNUMBER IN WORD 6 | 00029100 |
| FXTERNAL EVENTS ARE JOB ENTRY AND OUT OF SERVICE | 00029200 |
| PRIORITY ORDER IS END. OF. SIMULATION. CALL. END. | 00029300 |
| JOB.ENTRY, OUT.OF.SERVICE | 00029400 |
| | 00029450 |
| ACCUMULATE MO AS THE HEANING AS THE VARIANCE, AND | 00029500 |
| PROBQID TO 50 BY 1) AS THE HISTOGRAM OF N.QUEUE | 00029600 |
| ACCUMULATE UTILIZIO TO 6 BY 1) AS THE HISTOGRAM OF SWT | 00029700 |
| TALLY MC AS THE MEAN, VC. AS THE VARIANCE, AND HISTCIO TO 15 BY 1) | 00892000 |
| AS THE HISTOGRAM OF COUNT | 00029900 |
| TALLY MAY AS THE MEAN, VAT AS THE VARIANCE, NAT AS THE NUMBER AND | 00030000 |
| FREQ(0 TO 15 BY 1) AS THE HISTOGRAM OF JOB.RESPONSE.TIME | 00030100 |
| TALLY MWT AS THE MEAN, VWT AS THE VARIANCE, NWT AS THE NUMBER. | 00030200 |
| AND HWT (0 TO 240 BY 5) AS THE HISTOGRAM OF WAIT.TIME | 00030300 |
| TALLY MSRTI AS THE MEAN, VSRTI AS THE VARIANCE AND NSRTI AS THE | 00030400 |
| NUMBER OF PI-NBD, SPONSE | 00030500 |
| TALLY MORTE AS THE MEAN, VORTE AS THE VARIANCE AND NORTE AS THE | 00030600 |
| NUMBER OF P2+NBD.RSPONSE | 00030700 |
| TALLY MSRTE AS THE MEAN&VSRTE AS THE VARIANCE AND WSRTE AS THE | 00030800 |

NUMBER OF P3.NBD.RSPONSE TALLY MSRT4 AS THE MEAN + VSRT4 AS THE VARIANCE AND NSRT4 AS THE NUMBER OF P4.NBD.RSPONSE TALLY MORTS AS THE MEAN, VORTS AS THE VARIANCE AND NORTS AS THE NUMBER OF P5.NBD.RSPONSE ACCUMULATE MAY AS THE MEAN, VAV AS THE VARIANCE AND HISTAV (0 TO 12 BY 1) AS THE HISTOGRAM OF N.AVAILABLE DEFINE CALCULATE.DISTANCE AS A REAL FUNCTION WITH 4 ARGUMENTS END MAIN DEFINE SIM. LENGTH AND HEAL. TIME AS REAL VARIABLES RESERVE REGION.NAME(*) AND TITLE(*) AS 20 START NEW PAGE FOR I = 1 TO 20, READ TITLE(1) AS (20) A 4 FOR I = 1 TO 20, READ REGION, NAME(I) AS (20) A 4 READ SIM.LENGTH: MAX.SENT, MEAL.DURATION, AND TOUR.LENGTH WRITE AS " INPUT DATA FOR " FOR I= I TO 20, WRITE REGION.NAME(I) AS A 4 WRITE AS / SKIP 3 LINFS PRINT 1 LINE WITH SIM. LENGTH AS FOLLOWS SIMULATION LENGTH = ******* . ** HOURS SKIP 1 LINE READ N.BLOCK .N. NBD. N. CAR LET N.AVAILABLE=N.CAR PRINT 1 LINE WITH N.BLOCK N. NBD AND N. CAR AS FOLLOWS NO. OF RLOCKS = **** NO. OF NBDS. = **** NO. OF CARS = **** RESERVE ADJACENT. CARS (* +*) AS N.NBD BY N.CAR.AND SECTORS (* +*) AS N.CAR BY # CREATE EVERY BLOCK, NBD, CAR, AND P.CLASS(5) SKIP 1 OUTPUT LINE PRINT I LINE AS FOLLOWS BLOCK ID TAX NO. NBD ID FOR I=1 TO N.BLOCK, DO READ B.TAXNO(B).NBDID(B).XCORD(B).YCORD(B) USING UNIT 7 PRINT 1 LINE WITH B, TAXNO(B), NBDID(B), XCOBD(B), YCORD(B) THUS **** *** **** ** ***** LOOP START NEW PAGE FOR I= 1 TO 20 . WRITE TITLE(I) AS A 4 WRITE AS / SKIP 2 LINES PRINT 2 LINES AS FOLLOWS NBD ID SECTORS NUMBER OF ORDER IN WHICH SECT.CARS ADJ.CARS CARS ARE NOMINATED FOR I=1 TO N.NBD. DO READ NONBDONAME (N) ON SECTOR CARS (N) ON ADJACENT (N) FOR J=1 TO N.CAR, READ ADJACENT. CARS(N.J) PRINT 1 LINE WITH NONBDONAME (N) ON SECTOR. CARS (N). N. ADJACENT (N) AS FOLLOWS *** **** ** 488 FOR K=1 TO N.CAR.DO PRINT 1 LINE WITH ADJACENT. CARS(I.K) AS FOLLOWS *** LOOP 1.00P 00 FOR I=1 TO N.CAR READ C, CAR.NAME(C) .NUMB.SECTORS(C) .CENTHOID(C) .MEAL.TIME

11

-69-

READ C.CAR.NAME(C).NUMB.SECTORS(C).CENTHOID(C).MEAL.TIME RESERVE SECTORS(C.*) AS NUMB.SECTORS(C) FOR J=1 TO NUMB.SECTORS(C). READ SECTORS(C.J)

| FOR T=1 TO N.P.CLASS READ VELOCITY(I) | 00037100 |
|--|----------|
| RESERVE P-DURATION (*) AS 3 | 00037200 |
| FOR $I = 1$ TO 3 PEAD P-DURATION(I) | 00037300 |
| CREATE AN END.OF.SIMULATION | 00037400 |
| SCHEDULE THIS END. OF. SIMULATION IN SIM. LENGTH HOURS | 00037500 |
| START STMULATION | 00037600 |
| STOP | 00037700 |
| FND | 00037800 |
| | 00037900 |
| EVENT FOR END.OF.SIMULATION | 00038000 |
| DEFINE MOVATATINATOUTATOTINATOTOUT AS REAL VARIABLES | 00038100 |
| DEFINE X AS A REAL 1-DIMENSIONAL ARRAY | 00582000 |
| $DESERVE X(4) \Delta S 10$ | 00038300 |
| ROCOTE AVA, HE IS | 00038400 |
| START NEW PAGE | 00038500 |
| PRINT 2 DOUBLE LINES AS FOLLOWS | 00038600 |
| FRINT 2 DOORLE LINKS AS FOLLOWS | 00038700 |
| THE NEW YORK CITY-RAND INSTITUTE | 00038800 |
| THE MEN CONT CITE OND I VOIL OFF | 00038900 |
| POLICE PATROL STALLATION MODEL | 00039000 |
| UDITE AS HETMIN ATTAIN DESID TS FAR D | 00010100 |
| $\frac{1}{2} \frac{1}{2} \frac{1}$ | 00030200 |
| FUR L- L TO DOA MAINT REGIONALANELLI AS A 4 | 00039200 |
| | 00039230 |
| Shift I LIME β and β white still (1) as β (4) | 00039300 |
| FUR IT L IN T WRITE TITETIT AS A 4 | 00039400 |
| WALLE AS / | 00039450 |
| SATE S DUTFUL LINES | 00039300 |
| PRINT I LING AS FULLONS | 00039000 |
| CAR ACTIVIT SUMMART | 00039700 |
| | 00039900 |
| SKIP I DUPPUT LINE BDINT O DOUDLE INEE AS EQUIQUS | 00039900 |
| CAR NO CE LORS ON DATROL PESDONDANC MORKING | 00040000 |
| LAR NUL OF JOBS ON PAIROL RESPONDING. HORKING | 00040200 |
| THE IN SCHULC | 00040200 |
| | 00040400 |
| IN UUI UUI UF SERVICE | 00040400 |
| FUN NELLIU NICANDU | 00040500 |
| | 00040000 |
| ET TODIACOTALICANELTOTILIZANETTOTILIZANENT/TOTILICANENT/TTECOV | 00040700 |
| THIS I DONEL LINE WITH CARANAME REPRINGSCOLOGICAL TO ISECT. JUBORNY | 00040500 |
| | 00040900 |
| | 00041100 |
| | 00041200 |
| waa waxa uuxa afaana afaana afaana xfaana xfaana afaana . | 00041200 |
| | 00041300 |
| FOR $J=1$ 10 / 7 DU ADD UTLIZ(RYJ)/IME V 10 X(J) LOUP | 00041600 |
| | 00041500 |
| | 00003300 |
| AUD TIN TO TOTALT | 00041700 |
| | 0041800 |
| | 00041900 |
| FUR J=1 TO 7 DO LET $X(J) = X(J) / N CAR 100P$ | 00042000 |
| LEI IUIIN#IUIIN/N.CAR LEI TOIOUI=TOTOUY/N.CAR | 00042100 |
| SKIP I OUTPUI LINE | 00042200 |
| PRINT] DUULLE LINE WITH IJ+0J+X(1)+X(2)+X(3)+X(4)+X(5)+X(6)+ | 00042300 |
| IUIINEIUIUUTex (7) IHUS | 00047400 |
| <u>IUTALS 99999 99966 92999 92999 92998 82999 92999 82999 82999</u> | 00042500 |

00042600 # # # # # * *** 00042700 FOR I=2 TO N.P.CLASS. DO START NEW PAGE 00042800 FOR J= 1 TO 20, WRITE TITLE(J) AS A 4 00042900 WRITE AS / 00042950 SKIP 2 LINES 00043000 PRINT 1 LINE WITH I AS FOLLOWS 00043100 QUEUEING STATISTICS FOR PRIORITY CLASS ** 00043200 00043300 SKIP 2 OUTPUT LINES PRINT 2 L'INES WITH MQ(I) AND VQ(I) AS FOLLOWS 00043400 HISTOGRAM OF QUEUE-SIZE AVERAGE QUEUE-SIZE = +++.+++ 00043500 NO. IN QUEUE FREQUENCY VARIANCE *** 00043600 00043700 SKIP 1 OUTPUT LINE FOR J= 1 TO 30 . WITH PROBQ(1.J) GT 0. DO 00043800 PRINT 1 LINE WITH J-1 AND PROBQ(1+J)/TIME, V AS FOLLOWS 00043900 **##**: * *** 00044000 LOOP 00044100 00044200 00044300 SKIP 2 DUTPUT LINES 00044400 LET T=NWT(I) *MWT(I) LET NTOT=NRT(I) 00044500 IF NTOT=0 LET T=0 GO TO PRT 00044600 OTHERWISE LET T=T/NTOT 00044700 PRT: PRINT 3 DOUBLE LINES WITH NWT(I), MWT(I), VWT(I), NTOT, THUS 00044800 HISTOGRAM OF WAITING TIMES AVERAGE WAITING TIME OF THE *** 00044900 JOBS DELAYED= ***. *** 00045000 00045100 WAITING TIME FREQUENCY VARIANCE = ###. ### 00045200 (MINUTES) AVERAGE WAITING TIME OF ALL **** 00045300 JOBS DISP D = ***.** 00045400 FOR J=1 TO 24+ WITH HWT(I+J) GT 0+ DO 00045500 00045600 PRINT 1 LINE WITH 5+J AND HWT(I,J) THUS *** 00045700 < *** 00045800 LOOP SKIP 3 OUTPUT LINES 00045900 00046000 LET J=1 PRINT 3 LINES AS FOLLOWS 00046100 CURRENT QUEUE CONTENTS 00046200 TIME SINCE RECEIPT 00046300 POSITION (MINUTES) 00046400 SKIP 1 OUTPUT LINE 00046500 00046600 FOR EVFRY JOB IN QUEUE(I) . DO PRINT 1 LINE WITH J AND (TIME.V-ENTRY.TIME(JOB))+1440.0 THUS 00046700 00046800 *** *** ** 00046900 ADD 1 TO J 00047000 LOOP 00047100 LOOP 00047200 CALL RESULTS ##SIMULATION## 00047300 END ROUTINE FOR RESULTS 00047400 00047500 DEFINE MOVOT AS REAL VARIABLES BEGIN REPORT PRINTING FOR I=1 TO N.P. CLASS IN GROUPS OF 6 PER PAGE 00047600 BEGIN HEADING 00047700 00047800 FOR J= 1 TO 20+ WRITE TITLE(J) AS A 4 00047850 WRITE AS / 00047900 SKIP 2 LINES 00048000 PRINT 1 LINE AS FOLLOWS 00048100 RESPONSE TIME HISTOGRAMS 00048200 SKIP 2 OUTPUT LINES 00048300 END I HEADING !! PRINT 1 DOUBLE LINE WITH A GROUP OF I FIELDS THUS 00048400

LET XLOC(C)=XCORD(CENTROID(C))

LET YLOC(C)=YCOPD(CENTROID(C))

LOOP

(C)

10

SCHEDULE AN OUT OF SERVICE GIVEN C IN MEAL TIME HOURS

00036700

00036800

00036900

00037000

| | 1 | 14 14 14 14 14 14 14 14 14 14 14 14 14 1 | | |
|--------------------------|---------------------|---|------------------------------|---------------------|
| PRIORTTY CLASS | \$ \$ | ** ** | ** | 44 (10048300 |
| # # | | | | 00048600 |
| SKIP 1 OUTPUT LINE | 1 · · · · · | | | 00048700 |
| PRINT 2 LINES AS F | OLLOWS | | | 00048800 |
| RESPONSE TIME | | | | 00048900 |
| (MINUTES) | | | | 00049000 |
| FOR $J = 1$ TO 16 | • 00 | | | 00049100 |
| PHINT 1 DOUBLE LINE | WITH J AND A | GROUP OF FREAD | I.J. FIELDS | THUS 00049200 |
| <*** | *** | ** | ** | **** 00049300 |
| *** | | | | 00049400 |
| LOOP | | | | 00049500 |
| SKIP 1 OUTPUT LINE | | | | 00049600 |
| PRINT 1 DOUBLE LINE | WITH A GROUP | OF NRT(I) FIEL | DS THUS | 00049700 |
| NUMBER OF RESPONSES | *** | ** *** | **** | ***** 00049800 |
| *** | | | | 00049900 |
| PRINT 1 DOUBLE LINE | WITH A GROUP | OF MRT(I) FIEL | DS THUS | 00050000 |
| AVERAGE H.T. | ** | ** ** ** | ** ** | **,** 00050100 |
| ** ** | | | | 00050200 |
| PPINT 1 DOUBLE LINE | WITH A GROUP | OF VRT(I) FIEL | DS THUS | 00050300 |
| VARIANCE OF R.T | **.** | 44 444 44 | 444,44 | ***.** 00050400 |
| *** ** | | | • • • • • | 00050500 |
| END HOFDONTH | | | | 00050600 |
| CTART NEW PAGE | | | | 00050700 |
| END 1 1 10 20 . WP | TTE TITLE(T) | AS A 4 | | 00050800 |
| WUTTE AC / | A SE FARMARE | FO N T | | 00050850 |
| CRID DIVINC | | | | 00050030 |
| BRIT E LINES | AS FOLLOWS | | | 00050900 |
| PRINT 1 DOODSE LINC | | SI | NUARY OF DE | PONSES BYA0051100 |
| | | 50 | INTIANI UP ME. | A0051200 |
| NEIGHDUNDUNU | | | | 00051200 |
| SKIP 2 UDIPUT LINES | S AC COLLOWE | | | 00051300 |
| PRINT 2 DUALE LINE | S AS FULLUNS | | | |
| PRIURITE 1 | | TOTALS | | 00051500 |
| | 7 | NO AVE DE V | | 00051800 |
| | | | AR NU | AVG R1 0005100 |
| | VAR NU | AVG NI VAN | | 00051800 |
| TAD 1 TO NUMBER DA | | | | 00051900 |
| FUR JEI TU NANDU DU | NO-NEDTO / IN | CT NO-NOTO (N | I FT NA -NODS | 00032000 |
| LEI NIENSHIINUT LEI | NEENSKICU | LEI NJ=NSKINUJ | LFI N4=NSK | 5(0) 00052100 |
| LEI IMNIHNETNISTNA | | ******** | 17 PP / 11 | 10052200 |
| LEI MENIAMORII(U) +N | CTMSKIC(J) +NJ | *MORIO(0) +N4*MO | | |
| | JANZANZAVSKIC | UJ TNJ MJ MJ MJ MJ MJ MJ | (J) +N4 +N4 + Y: Upput of | KID(J) 00052400 |
| TELEO LEI MEU LI | | PRI UI | MFRWISE | 00052500 |
| LEJ M#M/I | | | | 00052600 |
| LET V#V/{[#1] | | | | 00052700 |
| IPRI PRINT 1 DUUBLE L | INE WITH NBD. | NAME (J) INI I MSRT | 1(J) VSRT1 | JI+N2+ 00052800 |
| MSRT2(J)+VSRT2(J)+N | 3+MSRT3(J) + VS | RT3(J) N4 MSRT5 | $(J) \bullet VSRT5(J)$ | •T•M•V 00052900 |
| THUS | | | | 00053000 |
| **** **** | \$ \$ \$ | **** ****** | *** | * **.*** *00053100 |
| * *** | **** | ** ****** | V / | 00053200 |
| LOOP | | ta da serie da serie Serie da serie da ser | | 00053300 |
| SKIP I OUTPUT LINE | | | | 00053400 |
| LET T=0 LET M=0 | LET V=D | | | 00053500 |
| FOR 1=1 TO 4 DO | | | | 00053600 |
| IF I EQ 4. LET | I = 5 REGARD | LESS | 21 | 00053700 |
| LET N=NRT(I) | ADD N T | 0 T | | 00053800 |
| ADD N#MRT(I) T | O M ADD N#N | #VRT(I) TO V | | 00053900 |
| L00P | | | | 00054000 |
| LET MEM/T LET | V=V/(T*T) | | | 00054100 |
| PRINT 1 DOUBLE LINE | WITH NRT(1) . | MRT(1) . VRT(1) . N | RT (2) . MRT (2) | . 00054200 |
| VRT (2) +NRT (3) +MRT (3 |) + VRT (3) + NRT (| 5) + MRT (5) + VRT (5 |) . T . M . V THUS | 00054309 |
| TOTALS | * ** *** | **** ** *** ** | *** | ****** *00054400 |

-72-

*.*** **** **.*** **.** ***** 00054500 SKIP 9 OUTPUT LINES 00054600 PRINT 2 LINES AS FOLLOWS 00054700 PICK-UP (PRIDRITY 4) 00054800 NBD NO. AVG RT VAR SKIP 1 OUTPUT LINE 00054900 00055000 FOR J=1 TO N.NBD DO 00055100 LET N5 = NSRT4(J) PRINT 1 LINE WITH NBD. NAME (J) +N5+MSRT4 (J) +VSRT4 (J) AS FOLLOWS 00055200 00055300 *** *** 00055400 LOOP SKIP 1 OUTPUT LINE 00055500 PRINT 1 LINE WITH NRT(4), MRT(4), VRT(4) THUS 00055600 TOTALS 00055700 **** ** *** *** 00055800 START NEW PAGE FOR I = I TO 20 . WRITE TITLE(I) AS 00055900 A 4 00056000 WRITE AS / 00056050 SKIP 2 LINES 00056100 PRINT I LINE AS FOLLOWS DISTRIBUTION OF CAR AVAILABILITY 00056200 00056300 SKIP ,2 OUTPUT LINES 00056400 PRINT, 2 LINES AS FOLLOWS 00056500 NO. OF CARS PER CENT 00056600 ON PATROL OF TIME 00056700 FOR 1=0 TO NOCAR DO 00056800 PRINT 1 LINE WITH I AND 100*HISTAV(I+1)/TIME.V THUS 00056900 44 **** 00057000 LOOP 00057100 SKIP I OUTPUT LINE PRINT 1 LINE WITH MAV THUS 00057200 00057300 AVERAGE NUMBER AVAILABLE = **. *** 00057400 PRINT I LINE WITH VAV THUS 00057500 VARIANCE =+++ +++ 00057600 START NEW PAGE 00057700 FOR IN 1 TO 20 . WRITE TITLE(I) AS A 4 00057800 WRITE AS / 00057850 SKIP 2 LINES 00057900 PHINT | LINE AS FOLLOWS 00058000 NUMBER OF PRECINCT CARS SENT TO CALLS 00058100 SKIP 2 OUTPUT LINES 00058200 PRINT 1 DOURLE LINE THUS 00058300 PHTORITY 0 1 2 3 4 5 8 9 00058400 10 11 12 AVERAGE 13 14 15 VARIANCE 00058500 WRITE AS /, H 1 00058600 FOR I=1 TO 16+WRITE HISTC(1+1) AS I 6 00058700 WRITE MC(1) AND VC(1) AS S 6,0(6,2),5 5,0(7,2),/ 00058800 FOR 1=2 TO N.P.CLASS. DO PRINT 1, DOUBLE LINE WITH I.HISTC([.1) HISTC([.2) HISTC([.3), 00058900 00059000 HISTC(I+4)+MC(I)+ AND VC(I) THUS 00059100 ** **** 00059200 *** *** 00059300 LOOP 00059400 STOP 00059500 ENO 00059600 00059700 EVENT FOR JOB ENTRY 00059800 DEFINE HOUR AND MOUR AS REAL VARIABLES 00059900 READ LOC+ IPR+HOUR AND MOUR 00060000 00060100

00060200

00060300

CREATE A JOB CALLED J LET ENTRY.TIME(J)=TIME.V

LET LOCATION(J)=LOC

-73-

| LET PRIORITY(J)=IPR | 00060400 |
|--|-----------|
| LET DURATION(J)=HDUR*60.0 + MOUR | 00060500 |
| LET STATUS (J) =0 | 00060600 |
| GO TO PR(IPR) | 00060700 |
| IPR(1) CALL DISPR1(J) RETURN | 00060800 |
| PR(2) CALL DISPR2(J) RETURN | 00060900 |
| PR(3) CALL DISPR3(J) RETURN | 00061000 |
| PR(4) CALL DISPR4(J) RETURN | 00061100 |
| PR(5) CAL DISPBS(1) RETURN | 00061200 |
| END | 00061300 |
| DUNTING TO DISPRI GIVEN A | 00001000 |
| SET SECT-NEDID (LOCATION (A)) | 00061500 |
| | 00061500 |
| FOR 141 TO NICARIDU VINOMINATE CARS VI | 00001000 |
| | 00001700 |
| IF SWIRT LE 3 AND SPIRT NE I TTCAT IS ELIGIBLE T | 00001000 |
| IF MARKER=0 LET SUT(R)=1 | 00061900 |
| LET MARKER=1 | 00062000 |
| GO TO CALCI | 00062100 |
| OTHERWISE IF MARKER = 1 LET SOT(K)=2 | 00062200 |
| LET MARKER=2 | 00062300 |
| GO TO CALCI | 00062400 |
| OTHERWISE IF MARKER = MAX.SENT GO OUT | 00062500 |
| OTHERWISE ADD 1 TO MARKER LET SOT(K)=3 | 00062600 |
| CALC1: IF SWT(K)=1 CALL INTERPOLATE.LOCATION GIVEN K AND ASSIGNMENT(K) | 00062700 |
| REGARDIESS | 00062800 |
| 1F SPT(K) GT 1 CALL PREFMPT(K) | 00062900 |
| REGARDLESS CALL ASSIGN (K 1) 1 FT SWT (K) =2 | 00063000 |
| IF I GT N.SECTOR.CARS(SECT) FT SWT(K)=3 | 00063100 |
| | 00063200 |
| | 00063300 |
| REGARDLESS | 00003300 |
| 100P | 00003400 |
| TOTAL IF MARKER = 0 ADD I TO OUTSIDE DISPATCH | 00063500 |
| DESTRUT JOB CALLED J | 00063600 |
| REGARDLESS LET COUNT(I) #MARKER | 00063700 |
| RETURN | 00063800 |
| END | 00063900 |
| ROUTINE TO DISPR2 GIVEN J | 00064000 |
| LET SECT=NBDID(LOCATION(J)) LET MARKER=0 | 00064100 |
| FOR IN1 TO NOCAR, DO PINOMINATE CARS !! | 00064200 |
| LET K=ADJACENT, CARS (SECT, I) | 00064300 |
| IF SPT(K)=0 + CAR IS ELIGIBLE + | 00064400 |
| IF MARKER=0 LET SOT(K)=1 LET MARKER=1 GO TO CALC2 | 00064500 |
| OTHERWISE | 00064600 |
| IF MARKER=1 AND I LE N.ADJACENT(SECT)+N.SECTOR.CARS(SECT) | 00064700 |
| LET SOT(K)=2 LET MARKER=2 | 00064800 |
| CALC2 IF SWT(K)=1 CALL INTERPOLATE. OCATION GIVEN K AND | 00064900 |
| ASSIGNMENT (K) | 00065000 |
| REGARDLESS CALL ASSIGN(K+J) (FT SWT(K)=2 | 00065100 |
| TE I GT N.SECTOR.CARS(SECT) LET SWT(K) =3 | 00065200 |
| | 00065300 |
| | 00065400 |
| PECADDI ESS | 00065400 |
| | 000000000 |
| TE NADREDHA ETIE I TH AUDURIDA DETUDA | 00005000 |
| IT MAKNER FOR COUNTION-HADVED BETUDN | 00005100 |
| DIMENWISE LEI CUUNI(2)#MARKEM HEIUHN | 00005500 |
| END | 00065900 |
| ROUTINE TO DISPR3 GIVEN J | 00066000 |
| IT SEND ANY AVAILABLE CAR II | 00066100 |
| LET SECT=NRDID(LOCATION(J)) | 00066200 |
| FOR I=1 TO N.CAR . DO | 00066300 |
| LET KOAD JACENT CADS (SECT 1) | 00066400 |
| LL I A-ADDROENT SCARD (SCOTT) | (10040100 |

IF SPT(K)=0 TICAR IS ELIGIBLETT LET SOT(K)=1 IF SWT(K)=1 CALL INTERPOLATE.LOCATION GIVEN K AND ASSIGNMENT (K) REGARDLESS CALL ASSIGN(K+J) LET SWT(K)=2 IF I GT N.SECTOR.CARS(SECT) LET SWT(K)=3. REGARDLESS LET COUNT(3)=1 RETURN OTHFRWISE LOOP FILE J IN QUEUE(3) RETURN END ROUTINE TO DISPRA GIVEN J IN PICK-UP JOB IN LET SECT = NBDID(LOCATION(J)) FOR T = 1 TO N.SECTOR.CARS(SECT), DO LET K = ADJACENT.CARS(SECT.I) VICAR IS AVAILABLE! IF SPT(K) = 0 LET XLOC(K) = XCORD(LOCATION(J)) LET YLOC(K) = YCORD(LOCATION(J)) CALL ASSIGN(K.J) LET SOT(K) = 1LET SWT(K) = 2 LET COUNT(4) = 1 RETURN DTHERWISE LOOP FILE J IN QUEUE(4) IND SECTOR CAR FREET RETURN END ROUTINE TO DISPRS GIVEN J ASSIGN JOB ONLY TO SECTOR CAR !! LET SECT=NBDID(LOCATION(J)) LET K=ADJACENT.CARS(SECT.I) 1F SPT(K)=0 ICAR IS ELIGIBLE !! IF SWT(K)=1 CALL INTERPOLATE, LOCATION GIVEN & AND ASSIGNMENT(K) REGARDLESS CALL ASSIGN(K, J) LET SWT(K) #2 LFT SUT(K)=1 LET COUNT (5) =1 RETURN OTHERWISE LOOP **NO SECTOR CAR FREE** FILE J IN QUEUE(5) RETURN END ROUTINE TO INTERPOLATE . LOCATION GIVEN K AND J ** CALLED FROM PREEMPT AND FROM JOB.ENTRY ** NORMALLY MODE IS REAL DEFINE LUSKSJORSPATROL RETURN AS INTEGER VARIABLES LET LJ=LOCATION(J) LET V=VELOCITY (PRIORITY (J)) LET XD=XCORD(LJ)-XLOC(K) LET YD=YCORD(LJ) - YLOC(K) LET T=(TIME,V - T.START,RESPONSE(K))+24 LET TX=ABS.F(XD)/V LET DELTET - TX IF DELT GT O LET XLOC(K)=XCORD(LJ) LET YLOC(K)=V*DELT*SIGN,F(YD)+YLOC(K) GO TO CHECK.IF.PATROL OTHERWISE LET XLOC(K)=V*(-DELT)*SIGN.F(XD)+XLOC(K)

-75-

| 00066500 |
|-----------|
| 00066600 |
| 00066700 |
| 00066800 |
| 00066900 |
| 00067000 |
| 00067100 |
| 00067200 |
| 00067300 |
| 00067400 |
| 00067500 |
| 00067600 |
| 00067700 |
| 00067800 |
| 00067900 |
| 00068000 |
| 00068100 |
| 00068200 |
| 00058300 |
| 00068400 |
| 00068500 |
| 00068600 |
| 00068700 |
| 000000000 |
| 00006900 |
| 00069000 |
| 00069200 |
| 00009200 |
| 00009300 |
| 00009400 |
| 00009500 |
| 00069700 |
| 00069800 |
| 00069900 |
| 00070000 |
| 00070100 |
| 00070200 |
| 00070300 |
| 00070400 |
| 00070500 |
| 00070600 |
| 00070700 |
| 00070800 |
| 00070900 |
| 00071000 |
| 00071100 |
| 00071200 |
| 00071300 |
| 00071400 |
| 00071500 |
| 00071600 |
| 00071700 |
| 00071800 |
| 00071900 |
| 00072000 |
| 00072100 |
| 00072200 |
| 00072300 |
| 00072500 |
| VVUIZDUU |

-7

-76-

| CHECK TF . PATROL | IF SWT(K) NE 1 RETURN | 00072600 |
|-------------------|---|------------------------|
| | OTHERWISE CALL CANCEL. SECTOR. RETURN GIVEN K | 00072700 |
| • | RETURN | 00072800 |
| FND | | 00072900 |
| | | 00073000 |
| POUTINE TO ASS | IGN(K+J)- | 00073100 |
| IFT SPT(K)=PRI | OHITY(J) | 00073200 |
| LET ASSIGNMENT | (K)=J | 00073300 |
| NOW SCHEDULE .C | AR. ARRIVAL (K.J.) | 00073400 |
| FILF K IN | ASSIGNED.CARS(J) | 00073500 |
| TF SWT (K) L | E 1 SUBTRACT 1 FROM N.AVAILABLE REGARDLESS | 00073600 |
| RETURN | | 00073700 |
| END | | 00073800 |
| | | 00073900 |
| ROUTINE TO CAN | CEL.SECTOR.RETURN GIVEN K | 00074000 |
| LET PATROL .RET | IJRN = ASSIGNMENT(K) | 00074100 |
| DESTROY JOB C | ALLED PATROL.RETURN | 00074200 |
| LET RENEXT.EVE | NT (K) | 00074300 |
| CANCEL RETURN. | TO.SECTOR CALLED R | 00074400 |
| HESTPOY RETURN. | TO.SECTOR CALLED R | 00074500 |
| RETURN | | 00074600 |
| END | | 00074700 |
| | | 00074800 |
| ROUTINE TO PRE | EMPT(K) | 00074900 |
| LET J = ASSIGN | MENT (K) | 00075000 |
| IF IS NOT IN | A QUEUE | 00075100 |
| | JIEU FILE U FIRST IN WEDELPRIURITTUTT | 00075200 |
| PEGARULESS | E- NEXT EVENITIEN CANCEL THE ADDIVAL AT SCENE CALLED | 00075300 E 00075400 |
| NECTONA THE AD | ES NEATSCRETENTIAL GANGEL THE ARRIVALSATSCENE LALLED | 00075500 |
| DEWOVE K EROM | ASSIGNED.CADS(1) LET GPT(K)=0 | 00075600 |
| NOW INTERPOLAT | FLIOCATION GIVEN K AND J | 00075700 |
| TE N.ASSIGNED. | CARS(J) =1 (ET F=F.ASSTGNED.CARS(J) | 00075800 |
| IF SOT(F)= | 2 AND SWT(F) GF 4 | 00075900 |
| LET F = NE | XT.EVENT(F) | 00076000 |
| CANCEL THE | CALL END CALLED E | 00076100 |
| RESCHEDULE | THE CALL END CALLED E IN DURATION (J) MINUTES | 00076200 |
| REGARDLESS | | 00076300 |
| LET SOT (F) | zi e se a construction de la constr | 00076400 |
| REGARDLESS | | 00076500 |
| LET T.START.RE | SPONSE(K) = TIME.V | 00076600 |
| RETURN | | 00076700 |
| END | | 00076800 |
| | | 00076900 |
| ROUTINE TO SCH | EDULE.CAR.ARRIVAL(K,J) | 00077000 |
| DEFINE DIST AS | A REAL VARIABLE | 00077100 |
| LET DIST#CALCU | LATE DISTANCE (XCORD (LOCATION (J)) + YCORD (LOCATION (J)) | • 00077200 |
| XLOC(K) • YLDC(K |)) set of the set o | 00077300 |
| LEI I.STARIARE | SPONSE (R) #11ME v | 00077400 |
| CREATE AN ARRI | ADDINAL AT SOCHE GIVEN K IN | 00077500 |
| SCHENULE THIS | ANKIVALERIEJULNE JIVLN N IN | 00011000 |
| LET NEVT EVENT | TRIURSTINUT AT SCENC | 00077800 |
| RETURN | IN/-MINISTREAMIGOULINE | 00077000 |
| ENC | | 00078000 |
| 6 4 T | | 00078100 |
| EVENT FOR ARRT | VAL .AT.SCENF (K) | 00078200 |
| DEFINE D AS A | REAL VARTABLE | 00078300 |
| LET LKAR=LOCAT | ION (ASSIGNMENT (K)) | 00078400 |
| LFT XLOC(K) =XC | OHD (LKAR) | 00078500 |
| LET YLOC(K)=YC | DRD (LKAR) | 00078600 |
| | | |

15 SWT (K) =2 LET SWT (K) =4 ADD 1 T DTHERWISE LET SWT(K)=5 ADD 1 T PROTHE LET J=ASSIGNMENT(K) LET JPR=PRIORITY(J) ET D = UURATION(J)TF SOT(K)=1 ! THIS IS THE PRIMARY OTHERWISE IF STATUS(J)=0 ++NO CAR HAS ARRIV GO TO FIRST.CAR OTHERWISE. IF STATUS(J)=1 + PRIMARY CAR HAS PERFORM END. CALL. SCHEDULING GIVEN REGARDLESS RETURN II IF NO PRIMARY FIRST.CAR! LET STATUS(J)=2 LET JOR . RESPONSE . TIME (JPR) = (TIME . V T.START.RESPONSE (K)) #1440.0 GO TO RT (JPR) +RT(1) + LET P1.NBD.RSPONSE(NUDID(LKAR IRT(2) + LET P2, NBD, RSPONSE (NBDID (LKAR IRT(3) . LET P3.NBD.RSPONSE (NBDID (LKAR "PT(4) + LET P4.NBD.RSPONSE(NBDID(LKAR PT(5) LET PS.NBD.RSPONSE (NHDID (LKAR PRIME CART IF STATUS(J) NE O "" SOME CARS ALRI FOR EACH CAR IN ASSIGNED. CARS PERFORM END. CALL. SCHEDULING G LOOP LET STATUS(J)=1 RETI OTHERWISE LET JOR, RESPONSE, TIME (JPR) = (TIME, V T.START.RESPONSE(K))#1440.0 CALL END. CALL. SCHEDULING GIVEN K A LET STATUS (J)=1 HO TO RT (JPR) ENI ROUTINE TO CALCULATE DISTANCE GIVEN DEFINE D.X1.X2.Y1. AND YZ AS REAL LET D = ARS.F(X2-X1) + ABS.F(Y2-Y1 RET. RN WITH U END ROUTINE FOR END.CALL.SCHEDULING GI DEFI E T AND DUR AS REAL VARIABLES LET T=P, DURATION (SOT (K)) +DUR CREATE & CALL-END SCHEDULE THIS CALL END GIVEN K IN RETURN FND EVENT CALL.FND GIVEN K IF SWT (K) = 6 GO TO CHECK Q FEND OTHERWISE LET J=ASSIGNMENT(K) REMOVE K FROM ASSIGNED. CARS(J) IF ASSIGNED.CARS(J) IS EMPTY DESTRO REGARDLESS ICHECK.QI FOR EACH JOB ON QUEUE(1) !!TH) WITH LOCATION (JOB) =K. DO LET A SCHEDULE & CALL-END GIVEN K IN DURA LET SPT(K)=6 LET SDT(K)=6

| 7 | |
|---|--|
| | |

| O IN-SECT.JOBS(K) GO TO BOTH | 00078700 |
|---|-----------|
| O OUT+SECT+JOBS(K) | 00078800 |
| | 00078900 |
| | 00079000 |
| | 00079100 |
| 040 AT 144 | 00079200 |
| CAR AT JOR JIT GO TO PRIME.CAR | 00079300 |
| ED AND THE CAP IS NOT ODIN PALE | 00079400 |
| ED AND THIS CAR IS NOT PRIMARTIT | 00079500 |
| | 00079600 |
| ALREADY ARRIVED | 00079700 |
| K AND D | 00079800 |
| CAR DELAY SCHEDULING & CALL END | 100080000 |
| | 00080100 |
| | 00080200 |
| | 00080300 |
| | 00080400 |
| | 00080500 |
|)) = JOH, RESPONSE, TIME (JPR) RETURN | 00080600 |
| DESPONSE TIME (JPR) RETURN | 00080700 |
| 1) = 106 PEEPONEE TIME (JPR) RETURN | 00080800 |
| 1) # 108 RESPONSE TIME (JPR) RETURN | 00080900 |
| TAGODERESPONSE TIME (UPR) RETURN | 00081000 |
| EADY THERE! | 00081300 |
| (J) WITH SWT (CAR) GF & DO | 00081200 |
| IVEN CAR AND D | 00081400 |
| URN | 00081500 |
| | 00081600 |
| | 00081700 |
| | 00081800 |
| NU U | 00081900 |
| | 00085000 |
| | 00082100 |
| | 00092200 |
| N XI YI X2 AND Y2 | 00002300 |
| VARIABLES | 00082500 |
| Y and the second sec | 00082600 |
| | 00082700 |
| | 00082800 |
| | 00082900 |
| VEN K AND DUR | 00083000 |
| | 00083100 |
| | 00083200 |
| TMINITES | 00083300 |
| | 00083400 |
| | 00083600 |
| | 00083700 |
| OF OUT-OF-SERVICE | 00083400 |
| | 00083900 |
| | 00084000 |
| DY JOH CALLED J | 00084100 |
| | 00084200 |
| | 00084300 |
| SETEMBENTINAL ING CONTRACTOR | 00084400 |
| ADDAMMAENTERTANIS LET SWITCH #6 | 00084500 |
| ALTONIOUN MINULO | 00004000 |
| | 00004700 |

| LET XLOC(K) #XCOHD(CENTROID(K)) | 00084800 |
|---|-----------------------|
| LET YLOC(K)=YCORD(CENTROID(K)) | 00084900 |
| REMOVE JOH FROM QUEUE(1) | 00085000 |
| RETURN | 00085100 |
| LOOP | 00085200 |
| FOR I=2 TO N.P.CLASS DO | 00085300 |
| IF QUFUE(I) IS EMPTY GO TO NEXT.Q | 00085400 |
| OTHERWISE IF I LE 3 NOW REASSIGN.CAR.TO.JOB(K.F.QUEUE(T)) RETUR | IN 00085500 |
| OTHERWISE IND A SECTOR CARI | 00085600 |
| FOR EACH JOB ON QUEUE(I) DO | 00085700 |
| LET NHOOD=NBDID(LOCATION(JOB)) | 00085800 |
| FOR J=1 TO NUMB.SECTORS(K) DO | 00085900 |
| IF NHOOD=SECTORS(K,J) | 00086000 |
| NOW REASSIGN, CAR. TO. JOB (K. JOB) | 00086100 |
| RETURN | 00086200 |
| OTHERWISE | 00086300 |
| LOOP | 00086400 |
| LOOP | 00086500 |
| NEXT-Q1 LOOP | 00086600 |
| REVENT.PATROL . | 00086700 |
| ADD 1 TO N.AVAILABLE | 00086800 |
| NOW PLACE CAR. ON PATROL (K) | 00086900 |
| RETURN | 00087000 |
| | 00087100 |
| | 00087200 |
| ROUTINE TO REASSIGN.CAR.TO.JOB GIVEN K AND J | 00087300 |
| LET SECT=NBDID(LOCATION(J)) | 00087400 |
| IF PRIORITY(J)=4 LET XLOC(K)=XCORD(LOCATION(J)) | 00087500 |
| LET YLOC(K) =YCORD(LOCATION(J)) REGARDLESS | 00087600 |
| NOW ASSIGN (K+J) | 00087700 |
| FOR T # 1 TO N-CAR DO | 00087800 |
| IFT KKEADJACENT.CARS(SECT.I) | 00087900 |
| IF KK NE K GO TO LOOP1 | 00088000 |
| OTHERWISE LET SWT(K) #2 | 00088100 |
| IF I GT N.SECTOR.CARS(SECT) FT SWT(K)=3 | 00088200 |
| REGARDIESS GO TO OUT | 00088300 |
| 00P11 | 00088400 |
| 1008 | 00088500 |
| | 00088600 |
| LET PRISPRIORITY (J) | 00088700 |
| REMOVE (FROM QUEUE (PRT) | 00088800 |
| (ET WAIT.TIME(PRT) = (TIME.V - ENTRY, TIME(J)) + 1440.0 | 00088900 |
| FT COUNT(PRI)=1 | 00089000 |
| 1 FT SOT(k) = 1 | 00089100 |
| RETURN | 00089200 |
| FND | 00089300 |
| ROUTINE TO PLACE CAR ON PATROL GIVEN K | 000000000 |
| OFFINE D AS A REAL VARIARIE | 0000000000 |
| IFT SPT/KIED IFT SOT/KIED | 60009500 600005000 |
| TE SUTINING OR SUTINING IFT SUTINIAN RETUDN - SCAD WAR TH CERT | 00009000 |
| AL READY | 00000100 |
| NTHERMICE | 00003000 |
| ADEATE A LINE CALLER DATON DETHIDAL | 00009900 |
| UNETIE A VUE GALLEU FAIRVLAREIUNN IET Setikiei ist i ierentententei irt i nenttani detunni - | 00090000 |
| LET ACCIONMENTIKIEDATON DETHON | 00030100 |
| LET BOJIONMENT (NJEMATO) BETUDNI - | 002040200 |
| LEI PRIVELIT (PATROLIKETURN) SS | 00090300 |
| LEI U = UALCULATE.DISTANCE (XCONU(LJ),YCORD(LJ),XLOC(K),YLOC(K)) | 00090400 |
| LEI I START KESPONSE (K) =TIME V | 00090500 |
| CREATE A RETURN, TO, SECTOR CALLED R | 00090600 |
| LET CARNUMBER (R) *K ITHE CAR NUMBER OF THIS PATROL.RETURN ** | 0 009070 0 |
| SCHEDULE THE RETURN. TO. SECTOR CALLED R IN D/VELOCITY (3) HOURS | 00090800 |

LET NEXT EVENT (K) #R RETURN END EVENT RETURN. TO. SECTOR GIVEN K LET J=ASSIGNMENT(K) LET XLOC(K)=XCORD(CENTROID(K)) LET YLOC(K)=YCORD(CENTROID(K)) LET SWT(K)=0 DESTROY JOB CALLED J DESTROY THIS RETURN. TO. SECTOR RETURN END EVENT OUT. OF. SERVICE GIVEN K SAVING THE EVENT NOTICE DEFINE TOOUT. OF. SERVICE AS A REAL VARIABLE LET X.MEAL = OUT.OF.SERVICE IF EUNIT.A (OUT. OF. SERVICE) EQ 0 SCHEDULE AN OUT. OF. SERVICE GIVEN K IN TOUR. LENGTH HOURS LET T.OUT.OF.SERVICE=MEAL.DURATION GO AROUND OTHERWISE READ K. T.OUT.OF.SERVICE AROUND DESTROY THE OUT. OF. SERVICE CALLED X. HEAL IF SWT(K) GE 2 GO TO DELAY OTHERWISE IF SWT(K)=1 CALL CANCEL.SECTOR.RETURN GIVEN K LET XLOC(K) = XCORD(CENTROID(K)) LET YLOC(K) =YCORD(CENTROID(K)) REGARDLESS SUBTRACT 1 FROM N.AVAILABLE LET SWT(K)=6 LET SPT(K)=6 LET SOT(K)=6 CREATE CALL .END SCHEDULE THIS CALL.END GIVEN K IN T.OUT.OF.SERVICE MINUTES RETURN IDELAY! FOR EVERY JOB ON QUEUE(1) DO IF LOCATION(JOB) =K GO TO ALREADY.SCHEDULED OTHERWISE LOOP CREATE JOB LET LOCATION (JOB) =K LET FNTRY.TIME(JOB) =TIME.V LET DURATION (JOR) = T.OUT. OF. SERVICE LET PRIORITY(JOB)=6 LET STATUS (JOB) #6 FILF JOB IN QUEUE(1) RETURN + ALRFADY - SCHEDULED+ ADD T.OUT. OF . SERVICE TO DURATION (JOB) RETURN END

-79-



-81-Appendix B SAMPLE CONTROL CARDS AND INITIALIZATION DECK FOR THE 71ST PRECINCT IN NEW YORK CITY

The following control cards are sufficient for running the police patrol simulation, assuming that (1) the compiled program has been named HPATROL; (2) the output is to be listed on logical unit 3; (3) the job stream is to be read from logical unit 4; (4) the initialization data (except for the BLOCK data) is to be

- read from logical unit 5;
- unit 7.

//GO EXEC PGM=HPATROL, REGION=150K //GO.SIMU03 DD SYSOUT=A, DCB=(RECFM=FBA, LRECL=133, BLK5IZE=1729) //GO.SIMU04 DD DISP=SHR, DSN=RA701.R9057.PD.SHIFT3, UNIT=ONLINE, // VOL=SER=RES306 //GO.SIMU05 DD DDNAME=SYSIN //GO.SIMU07 DD DISP=SHR, DSN=RA701.R9057.HBE.TXBL71, UNIT=ONLINE, // VOL=SER=RES333 //GQ.SIMUIT DD DISP=SHR,DSN=A001.SIM2.ERRLIB

(5) the initialization data for the BLOCKs is to be read from logical

-83-

The following represents the initialization data for the 305 BLOCKs in the simulation of Precinct 71. This is the SIMOU7 data set.

The following cards constitute the complete initialization deck, except for the BLOCK data, for the sample simulation of the 71st Precinct in New York City.

.

| | | | | | | | | | | | BTOCK ID | lax NO. | NRD T |
|-----|-----|---------------------------|--------------------|---------|-----------|--|------------|-------------|---------|--|--|---------|-------|
| | | | | | | | | · · · · · · | | Į | 1 | 1185 | 12 |
| | | | - | | | | | | , n the | | 2 | 1187 | 12 |
| | | GO.SY | SIN | DD # | | | | | | | 3 | 1188 | 12 |
| | • • | | 6 | PATROL | CARS AND | 1 SERGEAN | TIS CAR | | | | 4 | 1189 | 15 |
| | | | TH | E 7157 | PRECINCT. | BROOKLYN | , NEW YORK | | | | 5 | 1190 | 12 |
| | 1 | 6.0 | 3 | 60 | 8 | | | | | | 6 | 1192 | 12 |
| | | 305 | 12 | 7 | | | | | | n en | 7 | 1196 | 12 |
| | 1 | A 1 | | 1 | | | | | | | 8 | 1197 | 11 |
| | 1 | 23 | 45 | 6 7 | | | | | | | 9 | 1198 | 11 |
| | 2 | 8 1 | | 1 | | | | | | | 10 | 1266 | 15 |
| | 1 | 234 | 5 6 | 7 | | | | | | | 11 | 1267 | 12 |
| | . 3 | C | 1 | 1 | | | | | | | 12 | 1268 | 9 |
| | 2 | 3 4 5 | 6 1 | 7 | | | | | | | 13 | 1269 | -9 |
| | 4 | D - | 1 | 1 | | | | | | | 14 | 1270 | . 9 |
| | .2 | 3 4 5 | 6 1 | 7 | | | | | | | 15 | 1271 | 9 |
| | 5 | E 1 | | 1 | | | | | | | 16 | 1272 | |
| | 3 | 1 4 2 | 2 6 5 | 7 | | | | | | | 17 | 1213 | 12 |
| | 6 | F | 1 | 1 | | | | | | 1 Contraction of the second se | 18 | 12/4 | 12 |
| | 3 | 1 4 2 | 6 5 | | | | - | | | | 19 | 1275 | 9 |
| | 7 | G | 1 | 1 | | | | | - | | 20 | 12/0 | 9 |
| | 5 | 264 | -31 | | | | | | | | 22 | 1279 | 9 |
| | 8 | Н | | | | | | | | | 22 | 1270 | 9 |
| | | 632 | (, 🤉 1 | | | | | | | | 2.3 | 1217 | 12 |
| | 9 | 1 | | 1 7 | | | | | | | 25 | 1200 | 12 |
| | · • | 034 | 1 1 | . / | | | | | | | 26 | 1282 | 9 |
| | | $\frac{1}{2}$ | . 1. . 1. 1 | 1 7 | | | | | | | 27 | 1283 | 9 |
| | | ייס בי | + J] 1 | 1 | | | | | | | 28 | 1284 | ģ |
| | | | <u>י</u> ויבי כ | 1 · 7 | | | | | | | 29 | 1285 | 9 |
| | 1 | י יי כ אי כ | 1 | . 1 | | | | | | | 30 | 1286 | 6 |
| 1.1 | 5 | 5 4 2 | 2 3 1 | 7 | | | | | | | 31 | 1287 | 12 |
| | ĩ | AR . | ້ວ້ | 144 | 3.0 | | | | | | 32 | 1288 | 12 |
| | - î | 2 | 1 | • | | | | | | | 33 | 1289 | 8 |
| | 2 | cō | 2 | 222 | 4.0 | | | | | | 34 | 1290 | |
| | 3 | 4 | | | | | | | | | 35 | 1291 | 8 |
| | 3 | EF | 2 | 100 | 5.0 | | | | | | 36 | 1292 | . 8 |
| | 5 | 6 | | | | | | | | and and a second and | 37 | 1293 | 5 |
| | · 4 | HI | 2 | 43 | 3.0 | | | | | | 38 | 1294 | 15 |
| | 8 | - 9 | | | | | | | | and the second | 39 | 1295 | 15 |
| | 5 | JG | 2 | 296 | 4.0 | | | | | | 40 | 129601 | 8 |
| | 7 | 10 | | | | | | | | | 41 | 129602 | 8 |
| | 6 | KM | 2 | 63 | 5.0 | | | | | | 42 | 1297 | 8 |
| | 1 | 1 12 | | | | | | • | | | 43 | 1298 | 8 |
| | 7 | SGT | 1 | 59 3 | •5 | | | | | and a statement | 44 | 1299 | 8 |
| | 0 | | | | | • • | | | | | 45 | 1300 | 5 |
| | | 20.0 | 20 | .0 20 | .0 20.0 2 | 0.0 | | | | ana com | 46 | 1301 | 12 |
| | | 1.0 | 0.5 | 0.0 | | - | | | | an a | | | |
| | | | | | | and the second | | | | 7.1 · · · · | and the second | | |

-82-

ID

| <u> </u> | <u>Y</u> |
|------------|----------|
| 01170 | 2 19140 |
| +01110 ··· | 2 10140 |
| 67.9450 | 2.12950 |
| •76100 | 2.07780 |
| •76830 | 2.00850 |
| •75940 | 1,94630 |
| .7462.0 | 1.83550 |
| .73350 | 1.73110 |
| .69360 | 1,65230 |
| .71180 | 1.58130 |
| .96310 | 2.12540 |
| 1.08660 | 2.09430 |
| 1,21150 | 2.07400 |
| 1 25220 | 2 05/50 |
| 1 40100 | 2.03400 |
| 1.49190 | 2.03140 |
| 1.03190 | 2.01260 |
| 1.77100 | 1.98/20 |
| •94520 | 2.07420 |
| 1.07370 | 2.04180 |
| 1.20350 | 2,02170 |
| 1.34540 | 2.00060 |
| 1.48500 | 1.97950 |
| 1.62440 | 1,95770 |
| 1.76420 | 1.93680 |
| .92480 | 2.01610 |
| 1.05880 | 1.98270 |
| 1.19540 | 1.96240 |
| 1.33520 | 1.94160 |
| 1 47580 | 1 92010 |
| 1 41570 | 1 90770 |
| 1.017/0 | |
| 1.73460 | 1.07730 |
| 90450 | 1.95950 |
| 1.04540 | 1.92460 |
| 1.18560 | 1.90230 |
| 1.32660 | 1.88150 |
| 1.46690 | 1.86140 |
| 1.60610 | 1.83970 |
| 1.74740 | 1.81760 |
| .88370 | 1.90050 |
| 1.02960 | 1.86610 |
| 1.14210 | 1.84800 |
| 1.21210 | 1.83870 |
| 1.31790 | 1.82290 |
| 1.45730 | 1.79970 |
| 1.50720 | 1.78100 |
| 1 73730 | 1 76070 |
| 1+13/30 | 1 00000 |
| •8064U | 1.05550 |

| | | -84- | | | | | | -85 |
|-----------------|--------|--------------|-----------------------------|----------|--|----------|--------|-----|
| Tetower | Tax | Neigh- | Coord | inates | | | Tax | Nei |
| TD No | NO | Dor- | 77 | | | Internal | Block | bor |
| TD NO. | NO. | 1000 | <u> </u> | <u> </u> | | ID NO. | NO. | noo |
| 47 | 1302 | 12 | .87830 | 1.81010 | | 102 | 141501 | 5 |
| 48 | 130401 | 12 | •95710 | 1.79960 | | 103 | 141502 | 5 |
| 49 | 130402 | 12 | 1.00830 | 1.79570 | | 104 | 1417 | 5 |
| 50 | 130403 | 12 | 1.06330 | 1.79180 | | 105 | 1418 | 5 |
| 51 | 1305 | 8 | 1.16870 | 1.78460 | | 106 | 1419 | 5 |
| 52 | 130601 | 12 | •77060 | 1.73210 | | 107 | 142001 | 5 |
| 53 | 130602 | 12 | | 1.73100 | | 108 | 142002 | 5 |
| 54 | 1307 | 12 | 1.00270 | 1.73130 | | 109 | 1422 | 5 |
| 55 | 1308 | 8 | 1.16010 | 1./3150 | | 110 | 1423 | 5 |
| 56 | 130901 | 8 | 1.26380 | 1.75460 | | 111 | 1424 | 5 |
| 57 | 130902 | 8 | 1.33160 | 1.75960 | | 112 | 1425 | 5 |
| 58 | 130903 | 8 | 1.34150 | 1.72010 | | 113 | 1426 | 5 |
| 59 | 1310 | 8 | 1.45020 | 1.74000 | | 114 | 1427 | 5 |
| 60 | 1311 | 8 | 1.58850 | 1.72220 | | 115 | 1428 | . 4 |
| 61 | 1312 | 5 | 1.72900 | 1.70080 | | 116 | 1429 | 4 |
| 62 | 1313 | 11 | .82030 | 1.07640 | | 117 | 1430 | 4 |
| 63 | 1314 | 11 | •99810 | 1.67710 | | 118 | 3508 | 2 |
| 64 | 1315 | 11 | 1.15810 | 1.67760 | | 119 | 4588 | 4 |
| 65 | 1310 | 8 | 1.30580 | 1.67730 | | 120 | 458901 | 4 |
| 60 | 1317 | 8 | 1.44560 | 1.67720 | | 121 | 458902 | 4 |
| 61 | 1318 | 11 | .81620 | 1.02050 | | 122 | 4590 | 4 |
| 60 | 1319 | 11 | •99820 • • • • • • • • • | 1.02040 | | 123 | 4591 | 4 |
| 70 | 1320 | 11 | 1.15740 | 1.62750 | | 124 | 4592 | 4 |
| 70 | 1321 | 8 | 1.30670 | 1.02010 | | 125 | 4593 | 2 |
| 71 | 1372 | 8 . | 1.44200 | 1.62670 | | 126 | 4594 | 2 |
| 72 | 1323 | 0 | 1007100 | 1.64640 | | 127 | 4595 | 2 |
| 73 | 1324 | 0 | 1.00040 | 1.64070 | | 128 | 4596 | . 2 |
| 76 | 1325 | 8 | 1.01710 | 1.64270 | | 129 | 4597 | 2 |
| 76 | 1320 | D • • | 10/1030 | 1.67000 | | 130 | 4598 | 2 |
| 70 77 | 1321 | 11 | 01030 | 1.57200 | | 131 | 4599 | ? |
| 79 | 1320 | 11 | +99700 | 1.57200 | | 132 | 4600 | 2 |
| 70 | 1327 | 11 | 1.10770 | 1.57600 | | 133 | 4602 | 3 |
| 80 | 1330 | 4 | 1.430/00 | 1.57490 | | 134 | 4603 | 3 |
| 01 | 1331 | 4 | 1 1443000 | 1 57220 | | 135 | 4604 | 3 |
| 0.0 | 1332 | 4 | 1 71040 | 1.57/00 | | 136 | 4605 | : 3 |
| 02 | 1333 | 4 | 1.01240 | 1.04520 | | 137 | 4606 | 3 |
| 8 / | 1394 | 6 | 2 0/070 | 1 0/250 | | 138 | 4607 | - 3 |
| 95 | 1395 | 6 | 2 19520 | 1 02100 | | 139 | 4608 | . 3 |
| 86 | 1207 | 6 | 2 33E0V | 1 00050 | | 140 | 4609 | 2 |
| 97 | 1371 | 2 | 2 45520 | 1 700/0 | | 141 | 4610 | 2 |
| | 13001 | с 2 | 2 49030 | 1 25040 | |] 42 | 4611 | 2 |
| 80 | 130002 | 2 | 2 63700 | 1 803000 | | 143 | 4612 | 2 |
| 07 QA | 1/00 | С. Б | 1 00440 | 1 01200 | | 144 | 4613 | 2 |
| 70 | 1400 | 0 | 1.70440 | 1.71300 | | 145 | 4614 | 2 |

1.89240

1.87080

1.84980

1.81830

1.85610

1.83450

1.81340

1.78990

1.79710

1.77590

1.75460

2.04200 2.17770

2.31080 2.55820 1.89670 2.03370

2.03370 2.16860 2.31010 1.88650 2.02310 2.15970

2

92

-85-

156

Coordinates

| and the second sec | |
|--|----------|
| X | <u> </u> |
| 0.00010 | 1 77.00 |
| 2.28210 | 1.73600 |
| 2.35200 | 1.72440 |
| 1.87830 | 1.73790 |
| 2.01610 | 1,71720 |
| 2.15180 | 1.69550 |
| 2.28330 | 1.67640 |
| 2.35070 | 1.67000 |
| 1.86820 | 1.67910 |
| 2.00810 | 1.65790 |
| 2.14620 | 1,63810 |
| 2.25270 | 1.62550 |
| 1.85940 | 1.62390 |
| 2.00810 | 1.62390 |
| 1.84840 | 1.57600 |
| 1.99180 | 1.57620 |
| 2.13790 | 1.57670 |
| 2.65480 | 1.72790 |
| 2.13750 | 1.52400 |
| 2,08000 | 1.47750 |
| 2.16220 | 1.47830 |
| 2 1 2740 | 1 42010 |
| 2 22600 | 1 42710 |
| 2.2000 | 1 40140 |
| 2.28500 | 1.43570 |
| 2.29241 | 1.51300 |
| 2.33320 | 1.53740 |
| 2.37400 | 1.56140 |
| 2.941740 | 1.58400 |
| 2.45710 | 1.60800 |
| 2.50370 | 1.63550 |
| 2.55220 | 1.66030 |
| 2.60180 | 1.69050 |
| 2.13480 | 1.32890 |
| 2.18570 | 1.32870 |
| 2.23660 | 1.32850 |
| 2.28370 | 1.32860 |
| 2.33190 | 1.32860 |
| 2.38110 | 1.29030 |
| 2.41630 | 1.26550 |
| 2.38960 | 1.37600 |
| 2.42850 | 1.40290 |
| 2.46780 | 1.43170 |
| 2.50640 | 1.45760 |
| 2.54640 | 1.48570 |
| 2.58980 | 1.51540 |
| 2.63300 | 1.54640 |
| 2.68120 | 1.57960 |
| 2.06040 | 1,19530 |
| 200070 | 1.10600 |
| 2 202411 | 1 10570 |
| 2 22050 | 1 10500 |
| 2 37050 | 1 10404 |
| 2.3/950 | 1 20020 |
| 2.43130 | 1 16600 |
| 2.47560 | 1.10090 |
| 2.47160 | 1.25940 |
| 2.51090 | 1.28660 |

| | Tax | Neigh- | Coordin | ates | 2 | | | Tax |
|--------------|--------|---|---------------|-----------|---|---|----------|-------|
| Internal | Block | bor- | 0002011 | | | | Internal | Block |
| ID No. | No. | hood | <u> </u> | <u> </u> | | | ID No. | No. |
| 167 | 1620 | | 2 64020 | 1 21470 | | | 212 | |
| 177 | 402.0 | 1 | 2 60970 | 1.314/0 | | | 212 | 4795 |
| 170 | 4027 | 1 | 2 6 3 6 0 7 9 | 1.34240 | | | 213 | 4796 |
| 109 | 4030 | | 2.02040 | 1 20020 | | | 214 | 4797 |
| 100 | 40.31 | 1 | 2 2 0 1 1 0 0 | 1.42030 | | | 215 | 4798 |
| 101 | 4032 | 1 | 2.71040 | 1.42930 | | • | 216 | 4799 |
| 162 1 | 4633 | | 2.00500 | 1.45810 | | | 217 | 4800 |
| 163 | 46.34 | . | 2.08580 | 1.09510 | | | 218 | 4801 |
| 164 | 4635 | 3 | 2.13350 | 1.09600 | | | 219 | 4802 |
| 165 | 4636 | 3 | 2.18060 | 1.09610 | | | 220 | 4803 |
| 166 | 4637 | 3 | 2.23170 | 1.09560 | | | 221 | 4804 |
| 167 | 46.38 | 3 | 2.21910 | 1.09520 | | | 222 | 4805 |
| 168 | 4639 | 3 | 2.32750 | 1.09480 | | | 223 | 4806 |
| 169 | 4640 | 3 | 2.37960 | 1.09520 | | | 224 | 4808 |
| 170 | 4641 | 3 | 2.42650 | 1.09540 | | | 225 | 4809 |
| 171 | 4642 | 3 | 2.47390 | 1.09420 | | | 226 | 4810 |
| 172 | 4643 | 3 | 2.52340 | 1.09250 | | | 227 | 4811 |
| 1/3 | 4644 | | 2.55930 | 1.06370 | | | 228 | 4812 |
| 174 | 4645 | 1 | 2.55360 | 1.14280 | | | 229 | 4813 |
| 175 | 4646 | 1 | 2.59320 | 1.17030 | | | 230 | 4814 |
| 176 | 4647 | 1 | 2.63210 | 1.19680 | | | 231 | 4815 |
| 1// | 4648 | | 2.67720 | 1.21480 | | | 232 | 4818 |
| 178 | 464901 | 1 | 2.70030 | 1.26510 | | | 233 | 4819 |
| 179 | 464902 | 1 | 2.75010 | 1.20820 | | | 234 | 4820 |
| 180 | 465001 | 1 | C. 1.3410 | 1,30840 | | | 235 | 4823 |
| 181 | 465002 | 1 | 2.78450 | 1.21220 | | | 236 | 4824 |
| 182 | 465101 | 1 | 2.81440 | 1.29180 | | | 237 | 4825 |
| 183 | 465102 | | 2.77090 | 1.35630 | | | 238 | 4826 |
| 184 | 4652 | 1 | 2.85240 | 1.34150 | | | 239 | 4827 |
| 185 | 4653 | 3 | 2.08960 | 1.00080 | | | 240 | 4828 |
| 186 | 4654 | 3 | 2.13640 | .99880 | | | 241 | 4829 |
| 187 | 4055 | 3 | 2.18390 | .99980 | | | 242 | 4837 |
| 188 | 4656 | 3 | 2.2.3270 | 1.00240 | | | 243 | 4838 |
| 189 | 4657 | 3 | 2.28060 | 1.00200 | | | 244 | 4841 |
| 190 | 4658 | 3 | 2.32640 | 1.00180 | | | 245 | 4842 |
| 191 | 4659 | 3 | 2.3/120 | 1.00270 | | | 246 | 4843 |
| 192 | 4660 | 3 | 2.42540 | 1.00090 | | | 247 | 4844 |
| 193 | 4661 | .3 | 2.47250 | 1.00110 | | | 248 | 4845 |
| 194 | 4002 | 3 | 2.52480 | 1.00010 | | | 249 | 4846 |
| 195 | 4003 | 3 | 265/3/0 | 1.00490 | | | 250 | 4847 |
| 196 | 4004 | 3 | 2.00490 | •97480 | | | 251 | 4848 |
| 1,9,7 | 400001 | | 2.07020 | .90220 | | | 252 | 4849 |
| 100 | 400002 | н Ц аранан ал | 2.02100 | 1.01740 | | | 253 | 4850 |
| 199 | 400001 | 1 | 2.67840 | 1.01/40 | | | 254 | 4851 |
| 200 | 400002 | 1 | 2 72800 | 1.05030 | | | 255 | 4853 |
| 202 | 400701 | 1 | 2 47030 | 1 1105430 | | | 256 | 4854 |
| 2112 | 400/UC | 1 | 2 75000 | 1 10000 | | | 257 | 4855 |
| 2113 | 4000 | n an the second s | 2 70210 | 1 13/30 | | | 258 | 4856 |
| 204 | 4007 | 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 | 2 02560 | 1.12420 | | | 259 | 4857 |
| 203 | 4070 | | 2 97000 | 1 10500 | | | 260 | 4858 |
| 200 | 40/1 | 1 | 2 02000 | 1 33040 | in an | | 261 | 4859 |
| 200 | 4012 | | 2.72990 | 1 52100 | | | 262 | 4860 |
| 200 | 4171 | 4 A | 1 42010 | 1.02100 | | | 263 | 4861 |
| 217 | 4176 | α το | 1 643710 | 1.52000 | | | 264 | 4862 |
| 211 | 4173 | 5 4 | 1 71140 | 1.52220 | | | 265 | 4863 |
| ⊂ (1 | 4174 | 4 | 101100 | 1.02230 | | | 266 | 4864 |

-86-

-87-

Neigh-bor-

Coordinates

| bood | | v | v |
|--------------|-------------------|-----------|---------------------|
| <u>1100u</u> | • | <u> </u> | L |
| 4 | | 1.84880 | 1.52400 |
| 4 | | 1.30610 | 1.47270 |
| 4 | | 1.43910 | 1.47270 |
| 4 | | 1.56750 | 1.47420 |
| 4 | | 1.71310 | 1.47420 |
| 4 | | 1.84910 | 1.47710 |
| 4 | | 1.30760 | 1.42470 |
| 4 | | 1.43940 | 1.42650 |
| 4 | | 1.56870 | 1.42680 |
| 4 | | 1.71270 | 1.42630 |
| 4 | | 1.84950 | 1.42860 |
| 4 | | 1.99050 | 1.52530 |
| 4 | 2 - ²⁷ | 1.98920 | 1.47710 |
| 7 | | 1.30800 | 1.37620 |
| 7 | | 1.43000 | 1 37820 |
| 7 | | 1 56890 | 1 32550 |
| · 7 · | | 1 71060 | 1 32690 |
| / | | 1 9/900 | 1 32000 |
| · · · · · | | 1 20700 | 1.30000 |
| - | | 1.30790 | 1.32340 |
| | | 1.43950 | 1.32550 |
| 3 | | 1.84980 | 1.32960 |
| 7 | | 1.30950 | 1.26970 |
| 1 | | 1.44130 | 1.2/380 |
| 3 | | 1.84980 | 1.27700 |
| 3 | | 1.94300 | 1.32820 |
| 3 | | 1.99040 | 1.32930 |
| 3 | | 2.03970 | 1.32910 |
| 7 | | 1.30850 | 1.21740 |
| 7 | | 1,30810 | 1.16120 |
| . 7 | | 1.57740 | 1.19310 |
| 7 | | 1.30970 | 1.08990 |
| . 7 | | 1.47760 | 1.08990 |
| · 7 · | | 1.55720 | 1.09110 |
| 7 | | 1.60540 | 1.09110 |
| 7 | | 1.65270 | 1.09200 |
| 7 | | 1.70060 | 1.09150 |
| 7 | | 1.75080 | 1.09090 |
| 3 | | 1.80110 | 1.09230 |
| 3 | | 1.84890 | 1.09200 |
| 3 | | 1.89570 | 1.09350 |
| 3 | | 1.94980 | 1.09340 |
| | | 1.99180 | 1.09470 |
| 3 | | 2.03300 | 1.09520 |
| 7 | | 1.31070 | 99860 |
| 7 | | 1.40970 | 99920 |
| 7 | | 1.45760 | 00880 |
| · | | 1.50540 | 00200 |
| ۲ ۳ | | 1 667340 | 1 00000 |
| 7 | | 1 60/00 | 1.00000 |
| | | 1 4500470 | 00000 |
| | | 1 700200 | ● ファフロ() 】 00000 |
| · · · · | | 1.70000 | 1.00020 |
| 1 | | 1 700/0 | • 33280 |
| 3 | | 1.19960 | •99930 |
| 3 | | 1.84670 | .99940 |
| 3 | | 1.89380 | •99960 |
| | | | |

A

| App | pendi | Lx C |
|--------|-------|--------|
| SAMPLE | JOB | STREAM |

The following series of jobs represents a possible job stream for a 16-hour sample simulation of the 71st Precinct in New York City. There are 114 calls for service and 10 OUT.OF.SERVICE events.

| Event Name | Tin | ne of E | vent. | |
|--------------------|-----|------------|-------|--|
| | Day | Hour | Min. | |
| JOB.ENTRY | 0 | 0 | 04 | |
| JOB . ENTRY | 0 | 0 | 04 | |
| JOB ENTRY | 0 | 0 | 07 | |
| JOB . ENTRY | 0 | 0 | 12 | |
| JOB . ENTRY | 0 | - 0 | 20 | |
| JOB . ENTRY | 0 | 0 | 27 | |
| OUT.OF.SERVICE | 0 | 0 | 31 | |
| JOB . ENTRY | 0 | 0 | 38 | |
| JOB, ENTRY | 0 | 0 | 42 | |
| JOB ENTRY | 0 | 0 | 48 | |
| JOB . ENTRY | 0 | 0 | 49 | |
| JOB . ENTRY | 0 | 0 | 49 | |
| JOB . ENTRY | 0 | 0 | 51 | |
| JOB, ENTRY | 0 | 0 | 52 | |
| JOB ENTRY | 0 | 1 | 21 | |
| JOB . ENTRY | 0 | 1 | 26 | |
| JOB . ENTRY | · 0 | 1 | 33 | |
| JOB . ENTRY | . 0 | 1 | 35 | |
| JOB . ENTRY | 0 | 1 | 37 | |
| JOB ENTRY | 0 | 1 | 59 | |
| JOB ENTRY | 0 | 2 | 11 | |
| JOB . ENTRY | -0 | 2 | 15 | |
| JOB . ENTRY | 0 | 5 | 16 | |
| JOB ENTRY | -0 | 5 | 17 | |
| JOB . ENTRY | -0 | 2 | 35 | |
| JOB . ENTRY | 0 | · 2 | 40 | |
| JOB . ENTRY | 0 | 3 | 01 | |
| OUT . OF . SERVICE | 0 | 3 | 18 | |
| JOB . ENTRY | 0 | 3 | 22 | |
| JOB . ENTRY | 0 | 3 | 26 | |
| JOB.ENTRY | 0 | 3 | 50 | |

| | Tax | Neigh- | Coordin | ates |
|--------------------|--------|--------|----------------|---------|
| Internal ID No. | No. | hood | X | Y |
| 267 | 4865 | 3 | 1.94580 | ,99990 |
| 268 | 4866 | 3 | 1.99220 | 9994(|
| 269 | 4867 | 3 | 2.03990 | 1.00030 |
| 270 | 5024 | 11 | .63870 | 1.61740 |
| 271 | 5026 | îî | .66850 | 1.35430 |
| 272 | 5028 | ÎÎ | .81760 | 1.52210 |
| 273 | 5029 | 11 | .99770 | 1.52090 |
| 274 | 5030 | 11 | 1.15780 | 1.52170 |
| 275 | 5031 | 11 | .81750 | 1.47280 |
| 276 | 5032 | 11 | .99870 | 1.47350 |
| 277 | 5033 | 11 | 1.15870 | 1.47400 |
| 278 | 5034 | 11 | .81720 | 1.4251 |
| 279 | 5035 | 11 | .99890 | 1.42630 |
| 280 | 5036 | 11 | 1.15810 | 1.4274(|
| 281 | 5037 | 11 | .81650 | 1.37850 |
| 282 | 5038 | 11 | .99930 | 1.37790 |
| 283 | 5039 | 11 | 1.15890 | 1.37780 |
| 284 | 5042 | 10 | .81660 | 1.32430 |
| 285 | 5043 | 10 | .99990 | 1.32450 |
| 286 | 5044 | 1.0 | 1.15900 | 1.32360 |
| 287 | 5045 | 10 | .82090 | 1.27320 |
| 288 | 5046 | 10 | .99890 | 1.27260 |
| 289 | 5047 | 10 | 1.15910 | 1.26940 |
| 290 | 5048 | 10 | .82330 | 1.21900 |
| 291 | 5049 | 10 | .99980 | 1.21780 |
| 292 | 5050 | 10 | 1.15880 | 1.21700 |
| 293 | 5054 | 10 | .68120 | 1,14170 |
| 294 | 5055 | 10 | . 82170 | 1.16240 |
| 295 | 5056 | 10 | .99940 | 1.16290 |
| 296 | 5057 | 10 | 1,16060 | 1.1617 |
| 297 | 5062 | 10 | .66040 | 1.03930 |
| 298 | 5063 | 10 | .71380 | 1.06110 |
| 299 | 5064 | 10 | .82830 | 1.09330 |
| 300 | 5065 | 10 | 1.00090 | 1.09160 |
| 301 | 5066 | 10 | 1.16120 | 1.09070 |
| 302 | 508301 | 10 | .83560 | 1.01700 |
| 303 | 508302 | 10 | .83140 | .98100 |
| 304 | 5084 | 10 | 1.00040 | 1.00010 |
| 305 | 5085 | 10 | 1.16080 | 1.00074 |
| | | | | |

ndix C

-89-

| | | Dura | tion [‡] |
|--------------------|----------|------|-------------------|
| Block ^T | Priority | Hour | Min. |
| 110 | 1 | 01 | 02 * |
| 99 | . 3 | 0.0 | 05 * |
| 282 | 3 | 01 | 25 * |
| 118 | 3 | 0.0 | 03 * |
| 62 | 3 | 0.0 | 16 * |
| 51 | 3 | 0.0 | 18 * |
| 2 | | | 21 * |
| 38 | 3 | 00 | 10 * |
| 17 | 1 | 00 | 55 ¥ |
| 51 | 3 | 0.0 | 43 # |
| 19 | 3 | 0.0 | 38 * |
| 255 | 1 | 00 | 19 # |
| 274 | 3 | 0.0 | 52 * |
| 118 | 3 | 0.0 | 09 * |
| 290 | 3 | 00 | 08 * |
| 242 | 3 | 0.0 | 19 * |
| 161 | 3 | 0.0 | 40 * |
| 130 | 3 | 00 | 06 # |
| 24 | 3 | 0.0 | 06 * |
| 275 | 3 | 0.0 | 14 * |
| 89 | 3 | 01 | 06 * |
| 123 | 3 | 00 | 48 * |
| 116 | 3 | 0.0 | 58 * |
| 293 | 1 | 00 | 21 * |
| 91 | 3 | 00 | 04 # |
| 292 | 3 | 00 | 09 # |
| 118 | 3 | 0.0 | 12 * |
| 4 | | | 15 * |
| 61 | З | 0.0 | 10 * |
| 213 | 3 | 0.0 | 33 * |
| 31 | 3 | 0.0 | 40 # |
| - | • | | |

*

| | | Dav | Hour | Min | Block | Dani ami ta- | llaure | M | | | | Event Name | Tin | ne of H | Event | | |
|-----------------|-----------|------------|---------|------|----------------|----------------|--------|----------|---|---|---|---|---------------|---------|--------|---|-------------|
| | | | | | DIOCK | Friority | Hour | Min. | | | | | Day | Hour | Min. | Block | Príority |
| JOB ENTRY | | 0 | 3 | 55 | 21 | 3 | 0.0 | 26 | 4 | | | INR ENTRY | <u>^</u> | 16 | 1.6 | | |
| JUB ENTRY | | . 0 | 4 | 50 | 294 | 3 | 0.0 | 35 | # | | | | 0 | 10 | 40 | 89 | 3 |
| OUT OF SERV | ICE | 0 0 | 4 | 02 | 3 | | | 70 | # | | | | . 0 | 10 | 47 | 20 | 3 |
| JOH, ENTRY | | 0 | 4 | 12 | 155 | 3 | 0.0 | 54 | # | | | | 0 | 10 | 40 | 125 | 1 |
| JOB ENTRY | | 0 | 4 | 15 | 155 | 3 | 0.0 | 15 | * | | | | 0 | 10 | 52 | 89 | 1 |
| JOB ENTRY | | . 0 | . 4 | 19 | 294 | 3 | 00 | 09 | # 100 100 | | | | 0 | 11 | 00 | 86 | 3 |
| JOB .ENTRY | | 0 | 4 | 22 | 77 | 3 | 0.0 | 17 | * | | | | | 11 | 10 | 271 | 3 |
| JOB ENTRY | | 0 | . 4 . | 23 | 34 | 1 | 0.0 | 42 | # | | | | 0 | 11 | 13 | 10 | 3. |
| JOB ENTRY | | 0 | 4 | 27 | 84 | 3 | 0.0 | 05 | * | | | JUB ENTRY | 0 | 11 | 16 | 127 | 3 |
| JOB ENTRY | | 0 | 4 | - 39 | 67 | 3 | 0.0 | 38 | ង | | | JUBAENIRT | 0 | 11 | 18 | 162 | 3 |
| JOR ENTRY | | 0 | 4 | 41 | 300 | 3 | 0.0 | 16 | # | | | JUBENIRT | 0 | 11 | 28 | 297 | . 3 |
| JOB.ENTRY | | 0 | 4 | 47 | 301 | 3 | 0.0 | 36 | # | | | JOBENIRY | 0 | 11 | 43 | 146 | 3 |
| JOB ENTRY | | 0 | 4 | 58 | 285 | 3 | 00 | 15 | # | | | JUBENIRY | 0 | 11 | 45 | 104 | 3 |
| JOB ENTRY | | 0 | .4 | 58 | 260 | 3 1 1 | 0.0 | 13 | # | | | JOH ENTRY | 0 | 11 | 45 | 205 | 3 |
| JOB, ENTRY | | 0 | 5 | 06 | 17 | 3 | 0.0 | 06 | # | | | JOB.ENTRY | 0 | 11 | 51 | 146 | 3 |
| JOB . ENTRY | | 0 | . 5 | 10 | 301 | 3 | 0.0 | 06 | # | | | JOB ENTRY | - 0 | 11 | 53 | 271 | 3 |
| JOB . ENTRY | | 0 | 5 | 10 | 218 | 3 | 0.0 | 08 | # | | | JOB . ENTRY | 0 | 12 | 16 | 243 | 3 |
| JOB . ENTRY | | 0 | 5 | 11 | 218 | 3 | 02 | 06 | * | | | OUT.OF.SERVICE | , Q | 12 | 18 | 1 | |
| JOB. ENTRY | | 0 | 5 | 22 | 34 | 1 | 0.0 | 20 | * | | | JOB . ENTRY | 0 | 12 | 31 | 284 | 3 |
| JOR ENTRY | | 0 | 5 | .30 | 16 | 1 | 0.0 | 08 | * | | | JOB.ENTRY | ., O : | 12 | 47 | 253 | 3 |
| JOB . ENTRY | | 0 | 5 | 34 | 290 | 3 | 0.0 | 55 | * | | | JOB . ENTRY | 0. | 12 | 58 | 221 | 3 |
| JOB ENTRY | | 0 | 5 | 36 | 145 | 3 | 0.0 | 03 | # | | | JOB . ENTRY | 0. | 15 | 59 | 134 | 3 |
| JOB ENTRY | | 0 | 5 | 45 | 126 | 3 | 0.0 | 12 | # | | | JOB . ENTRY | 0 | 13 | 01 | 225 | 3 |
| JOB ENTRY | | Ő | 5 | 51 | 161 | 3 | 0.0 | 07 | * | and the second se | | JOB . ENTRY | 0 | 13 | 11 | 242 | - 7 · |
| JOB ENTRY | | 0 | 6 | 01 | 241 | Ä | 00 | 16 | * | | | JOB ENTRY | 0 | 13 | 17 | 76 | . |
| JOB FNTRY | | 0 | 6 | 32 | 69 | 3 | ñň | 19 | 4 | r M | | JOB . ENTRY | 0 | 13 | 23 | 129 | · |
| JOB FNTRY | | Ň | 6 | 37 | 241 | 3 | 01 | ne Ne | 8 | | | JOB ENTRY | 0 | 13 | 33 | 162 | 3 |
| JOB FNTRY | | 0 | 6 | 51 | 103 | ີ ຈ ີ. | . 00 | 01 | 8 | | | JOB . ENTRY | 0 | 13 | 37 | 57 | 3 |
| IOB ENTRY | | 0 | 7 | 11 | 90 | . | 00 | - nà | 8. | | | JOB FNTRY | 0 | 13 | 39 | 104 | ່. ກໍ |
| OUT OF SERV | TCE | 0 | | 13 | 2 1 ··· | . . | 0.0 | 4.3 | ~ | | | JOB - ENTRY | 0 | 13 | 40 | 120 | . |
| IOP ENTRY | | 0 | <u></u> | 15 | 1 | • | 0.0 | 40 | े. स | | | OUT OF SERVICE | 0 | 13 | 42 | 6 | 3 |
| IOP ENTRY | | 0 | | 19 | 15 | 1 1 | 00 | 21 | 8 | | | JOB ENTRY | 0 | 13 | 45 | 279 | |
| JUD ENTRY | | , v | | 25 | 10 | 3 | 0.0 | 30 | | | | IOB ENTRY | ۰ ۵ | 14 | 21 | 10 | |
| JOB ENTRY | | 0 | 1 | 05 | 135 | 3 (1.6) | 00 | 30 | * | | | IOR ENTRY | ő | 14 | . 31 | 122 | 3 |
| JOD ENTRY | ter e tra | 0 | . H. | 00 | 11 | 3 | 0.0 | 16 | | | | IOR ENTRY | ň | 14 | 31 | 100 | 3 |
| JOD ENTRY | | 0 | 8 | 12 | 13 | 3 | 00 | 25 | н ж | | | | 0 | 14 | 72 | 299 | 3 |
| JUD ENTRY | | · · () | 8 | 24 | 126 | 3 | 01 | 2 D | ਸ . | | | | | 14 | 40 | 100 | 3 |
| JUB ENTRY | | 0 | 8 | 24 | 90 | 3 | 01 | 203 | 17 X | | | | 0 | 14 | 47 | 203 | . 3 |
| JUB ENTRY | | ·· 0 | .8 | 20 | 156 | 3 | 00. | 31 | F | - | | JUB ENTRY | 0 | 15 | 0.00 | 281 | 1 |
| JUB ENTRY | | 0 | 8 | 34 | 293 | 3 | 00 | . 13 | 8 | | | JUB ENTRY | 0 | 15 | 0.9 | 57 | 3 |
| JOB ENTRY | | . 0 | 8 | 35 | 291 | 3 | 0.0 | 10 | P | | | JUB ENTRY | 0 | 15. | 09 | 253 | 3 |
| JOH . ENTRY | · . | 0 | 8 | 40 | 118 | 3 | 00 | 1/ | ₽. | | : ' '. | OUT OF SERVICE | U. | 15 | 14 | 7 | |
| JOB . ENTRY | | 0 | 8 | 40 | 125 | 3 | 01 | 00 0 | ? | . If | | JOBCENTRY | 0 | 15 - | 18 | 242 | 3 |
| JOB . ENTRY | | 0 | 8 | 42 | 86 | 3 | 0.0 | 18 - | \$ | | | JOB ENTRY | 0 | 15 | 23 | 101 | |
| JOB . ENTRY | | 0 | 91 | 04 | 166 | 3 | 0.0 | 23 | F | | | JOB ENTRY | Ň | 15 | 23 | 191 | 3 |
| OUT.OF.SERV | ICE | . 0 | 9 | 28 | . 4 | | | 19 1 | | | e de la companya de l La companya de la comp | JOB FNTRY | ň | 16 | 10 | 107 | 3 |
| JOB . ENTRY | | · . 0 | 9 | 30 | 15 | 3 | 01 | 22 | A second sec second second sec | | | · • • • • • • • • • • • • • • • • • • • | a U | 1 (7 | IU | 1 (6 | 3 |
| JOB. ENTRY | | 0 | 9 | 42 | 62 | 3 | 0.0 | 50 | | | | | | | | | |
| OUT . OF . SERV | ICE | Ŭ, | 9 | 57 | ς. | 4 | | 38 | F | 0 | | For OUT.OF.SE | RVICE | events | this f | ield cont | ains the nu |
| IND ENTRY | | n | 10 | 11 | 202 | | 01 | 28 | • | | | to be placed out of | servi | ce. | | 14 - 14 - 14 - 14 - 14 - 14 - 14 - 14 - | |
| JUDELNIKT | | | 1.0 | | | | · · · | | | | | | | | | | |

-91-

Hour Min. 01 * 24 * 02 * 15 * 07 * 01 * 00 00 0.0 01 00 0.0 38 * 0.0 55 * 09 * 0.0 0.0 000000 18 * 08 * 54 * 10 * 18 * 32 * 0.0 0.0 00 01 * 60 * 18 * 54 * 0.0 00 11 * 30 * 00 0.0 0.0 08 * 01 02 # 00 08 * 08 * 32 * 09 * 06 * 22 * 30 * 02 * 09 * 02 * 09 * 02 * 09 * 21 * 00 00 00 00 00 00 00 00 0.0 0.0 40 * 0.0 12 * 09 * 03 * 0.0 0.0 32 * 00

Duration

number of the car

the length of time that the car is to remain out of service (in minutes).

-93-

Appendix D

A DEFINITION OF EACH GLOBAL SIMULATION VARIABLE

I. ENTITIES AND THEIR ATTRIBUTES

| Entity | Туре | Attribute | | | |
|--------|-----------|---------------|--|--|--|
| BLOCK | Permanent | TAXNO | | | |
| | | WOODD | | | |
| | | XCORD | | | |
| | | YCORD | | | |
| | | NBDID | | | |
| | | | | | |
| NBD | Permanent | NBD.NAME | | | |
| | | N. ADJACENT | | | |
| | | | | | |
| | | N.SECTOR.CARS | | | |
| | | P.NBD.RSPONSE | | | |

| CAR | Permanent | CAR.NAME | | |
|--|-----------|--------------|--|--|
| | | XLOC | | |
| | | YLOC | | |
| | | NUMB.SECTORS | | |
| | | | | |
| an a | | ASSIGNMENT | | |

T.START.RESPONSE

NEXT. EVENT

| <u>Mode</u> * | Description |
|---------------|--|
| I | The external identification number associated with the block. |
| R | The x coordinate of the center of the block. |
| R | The y coordinate of the center of the block. |
| I | The internal reference number of the neighborhcod to which the block belongs. |
| A | The 4-character name associated with the neighborhood. |
| I | The number of cars designated as adjacent resources for the |
| | neighborhood. |
| Ι | The number of sector cars assigned to the neighborhood. |
| R | The time until the arrival of the first car at an incident of priority n(n=1,,5) that occurred in the neighborhood (used as statistics-gathering |
| A | The 4-character name associated |
| R | with the car. The x coordinate of the car's |
| R | current location. The v coordinate of the car's |
| ••• - | current location. |
| 1 | to which this car is assigned as a sector car. |
| I | The internal identification number of the job on which the |
| D | car is currently working (if any). |
| ĸ | began responding to its cur- |
| I | The internal identification number of the next event that is scheduled to occur for |
| | the car. |

| | Entity · | Туре | Attribute | Mode* | Description | | Entity | Туре | At | <u>tribute</u> | Mode* | Description |
|---|---|---|--|------------|----------------------------------|--|----------------------|-----------|------------------|---------------------------------------|--|---------------------------------------|
| | | | SPT | I | The priority of the job cur- | | JOB | Tempor | ary L(| OCATION | I | The internal block number with- |
| | | | | | rently being serviced by the | | | | · · · · | · · · · · · · · · · · · · · · · · · · | | in which the job occurs. |
| | | | | | car (out-of-service jobs | | | | EN | TRY.TIME | ĸ | The time at which the call is |
| | | | | | have Priority 6; if the car | | | | | | | received (in days since the |
| | | | | | is available, its SPT is zero). | | | | | | | start of the simulation). |
| | | | - SOT | I | = 1 if the unit is working as | | | - - | PI | RIORITY | I | The priority of the job |
| | | | | | primary car. | | | | | | | (1-5). |
| | | | | | = 2 if the unit is working as | | | | S | TATUS | I | = 0: No car has yet arrived |
| | | | | | backup car. | | | | | | | at the scene. |
| | | | | | = 3 if the unit is working as | | | | | | | = 1: The primary car has |
| | | | | | a tertiary car. | | | | | | | already arrived at t-e |
| | | | SWI | I | = 0 if the unit is on patrol | | | | | | | scene. |
| | | | | | in its sector. | | | | | | | = 2: Some cars, but not the |
| | | | | | = 1 if the unit is returning | | | | | | | primary car, have already |
| | | | | | to sector from an outside | | | | | | | arrived at the scene. |
| | | | | | call. | | | | DI | JRATION | R | The length of time the primary |
| | | 1 | | | = 2 if the unit is responding | | | | | | | car is required to work at |
| | | | | | to a call in sector. | | | | | | | the job (in minutes). |
| | | | | | = 3 if the unit is responding | | | | | | | |
| | | | | | to a call outside of | , | | · . · · · | | | | |
| | | | | | sector. | | | | | | TT TTCTC | |
| | | | | | = 4 if the unit is working | | | | | | TT. 11010 | |
| | | | | | on a call in sector, | | | | Owner | Member | | |
| | | | | | = 5 if the unit is working on | | Name | | Entity | Entity | Tvpe | Definition |
| | | | | | a call out of sector. | | | | | | | |
| | | | and the second sec | | = 6 if the unit is out of | | OUEUE | | P.CLASS | JOB | FIFO | Jobs waiting to be dispatched. |
| | | | · · · · · · · · · · · · · · · · · · · | _ | service. | | ASSIGNE | D.CARS | JOB | CAR | FIFO | The cars assigned to a job. |
| 3 | | | CENTROID | I | The block to which the unit | | | | | | | |
| | | | | | goes when it returns to patrol | | | | | | 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1. 1 | |
| | | | | | in its sector after responding | | | | | | | |
| | | | TH CHOW TODE | - | to a job outside its sector. | | | | | | | |
| | | | TN. SECT. JUBS | T. | the total number of sector jobs | | | | | | | |
| | | | | - - | The total number of out of | | | | | | | |
| | | | 001*2701*2022 | . | The Local number of out-of- | | | | | | | |
| | | | | | spondod to co far | | | | | | | |
| | | • | | | sponded to so far. | | | | | | | |
| | P. CLASS | Permanent | TOB RESPONSE TTN | E R | The time until the arrival of | | | | | | | |
| | r.onnoo | I CIMMUCIIC | DOD & REDI ORDE & LAL | | the first car at a given inci- | | | | | | | |
| | | | an a | | dent of this priority class | | | | | | | |
| | | | | | (used as a statistics-gathering | a ang a sing a sin | | | | · | | |
| | | | | | variable). | | | | | • | | |
| | | | VELOCITY | R | The average speed at which cars | | | | | | | |
| | | | | | respond to incidents of this | | | | | | | |
| | | | | | priority class. | | | | | | | |
| | | e de la composición d En este de la composición de la composic | COUNT | I | The number of cars actually dis- | | | | | | | |
| | | | | | patched to a specific incident | | | | | | | |
| | | | an a | | of this priority class (a | | | * | wmbol c | need in th | is column has | ve the following meaning: |
| | | | | | statistics-gathering variable). | | | The S | улпота | T Tn | teger | · · · · · · · · · · · · · · · · · · · |
| | and a start of the second s | | WAIT.TIME | R | The time that a specific inci- | | | | | R Ro | a1 | |
| | | | | | dent of this priority class | | | | | A. A1 | phanumeric | |
| | | | | | spends waiting in queue | | | | n an Thursday | Alah | | |
| | | | | | before a unit is dispatched | | د. مرجعه محمد الم | | | | | |
| | $\mathcal{F}_{\mathcal{F}}$ | | | | to it (a statistics-gathering | | | | | | | |
| | an an Araba An Araba an Araba | | | | variable). | | | | | an an tha th | | |

-94-

-95-

Variable

N.AVAILABLE

OUTSIDE, DISPATCH

MQ, VQ, PROBQ UTILIZ MC, VC, HISTC MRT, VRT, NRT, FREQ MSRT_n, VSRT_n, NSRT_n, MAV, VAV, HISTAV

dispatch. is sent. N.AVAILABLE.

III. ARRAYS

-96-

| Name | No.of Dimen- sions | lst Dimen- sion | 2nd Dimension | |
|---------------|--------------------------|-----------------------|--------------------|---|
| ADJACENT.CARS | 2 ` | N.NBD | N. CAR | The dispatch nomination |
| SECTORS | 2 | N.CAR | NUMB.SECTORS (CAR) | The sector responsibilities for each car. |
| P.DURATION | 1 | 3 | | The fraction of a job's duration that the primary |
| | | | | car, secondary car, and tertiary car spend working |
| REGION.NAME | 1 | 20 | | The alphanumeric name of the region being simulated |
| TITLE | 1 | 20 | | (maximum 80 characters). A title identifying the simulation run (maximum 80 |
| | | | | characters). The title is printed at the top of each |

4

12

IV. SYSTEM VARIABLES - INPUT

output report.

| Variable | Mode | Description |
|---------------|------|--|
| N.BLOCK | I. | The total number of BLOCKs in the region being simulated. |
| N.NBD | Ĩ | The total number of NBDs in the region being simulated. |
| N.CAR | I | The number of cars being simulated. |
| N.P.CLASS | I | The number of priority classes (set at 5 in the simulation docmented here). |
| MAX, SENT | I | The maximum number of cars that can be dis- patched to a Priority 1 incident. |
| MEAL.DURATION | I | The length of an internally scheduled OUT.OF. SERVICE job. |
| TOUR.LENGTH | I | The length of each car's work day. |

V. SYSTEM VARIABLES - INTERNAL AND OUTPUT

Description

The current number of cars available for

The number of times that no car in the region being simulated is available for dispatch to a Priority 1 call for service. It is assumed that a car from outside the region

The mean, variance, and histogram of the queue lengths for each of the QUEUEs. The histogram of the amount of simulated time a car spends at each value of SWT. The mean, variance, and histogram of COUNT, the number of cars sent to an incident. The mean, variance, number, and histogram of JOB.RESPONSE.TIME.

The mean, variance, and number of P.NBD. RESPONSE (n=1, 2,..., 5). The mean, variance, and histogram of

-97-

REFERENCES

- Crabill, T. B., W. E. Walker, and P. Kolesar, "Validation of a Police Patrol Simulation Model," Proceedings of the 8th Annual Simulation Symposium, Tampa, Florida, March 1975.
- Fishman, George S., <u>Concepts and Methods in Discrete Event Digital</u> <u>Simulation</u>, John Wiley & Sons, Inc., New York, 1973.
- 3. Gordon, Geoffrey, System Simulation, Prentice-Hall, Inc., Englewood Cliffs, 1969.
- Ignall, E., P. Kolesar, and W. Walker, "The Use of Simulation in the Development and Empirical Validation of Analytical Models for Emergency Services," Proceedings of the 1974 Winter Simulation Conference, Washington, D.C., January 1974.
- Johnson, G. D., <u>SIMSCRIPT II.5 User's Manual S/360-370 Version</u>, Consolidáted Analysis Centers, Inc., Santa Monica, California, 1972.
- 6. Kiviat, P. J., R. Villanueva, and H. M. Markowitz, <u>SIMSCRIPT II.5</u> <u>Programming Language</u>, Consolidated Analysis Centers, Inc., Santa Monica, California, 1973.
- Kolesar, P., and W. E. Walker, "A Simulation Model of Police Patrol Operations: Executive Summary," R-1665/1-HUD, The New York City-Rand Institute, forthcoming.
- 8. Naylor, T. H., J. C. Balintfy, D. S. Burdick, and K. Chu, <u>Computer</u> <u>Simulation Techniques</u>, John Wiley & Sons, Inc., New York, 1966.

END