GUIDELINES FOR SCHEDULING POLICE PATROL CARS

PETER J. KOLESAR, KENNETH L. RIDER, THOMAS B. CRABILL

R-1803-NYC
DECEMBER 1976

THE
NEW YORK CITY
RAND
INSTITUTE

Rand
SANTA MONICA, CA. 90406

GUIDELINES FOR SCHEDULING POLICE PATROL CARS

PETER J. KOLESAR, KENNETH L. RIDER, THOMAS B. CRABILL

R-1803-NYC
DECEMBER 1976

THE
NEW YORK CITY
RAND
INSTITUTE

Rand
SANTA MONICA, CA. 90406

PREFACE

This report describes a method of generating patrol car schedules for use by police department systems analysts and management personnel. Various measures for evaluating schedules are described, and some general conclusions regarding the best schedules to use are discussed. This methodology was originally developed under a contract with the New York City Police Department to improve the deployment of patrol cars.

This work was essentially completed in August 1975, when the authors were affiliated with the New York City-Rand Institute.

SUMMARY

This report is a part of the general effort in recent years to develop quantitative methods to assist police departments in allocating resources and making manpower-scheduling decisions. It presents a methodology for creating new patrol car schedules that improve the correspondence between patrol car availability and demands for service.

The level of demand for police service varies considerably through the day. Recognizing this, police departments assign more patrol cars to duty during the busy hours. Scheduling decisions, however, have usually been educated guesses. Improvements resulting from better schedules can be quite significant. For example, a schedule devised using the methods presented here maintained performance standards with almost a 25 percent reduction in the number of cars fielded under the traditional schedule.

This report demonstrates the use of mathematical models to determine the hourly requirements for patrol cars. These requirements are input to a scheduling model that generates schedules that meet the requirements. The resulting schedules are then evaluated with respect to various measures of patrol system performance. Data from one of New York City's police precincts is used to show how the method was applied to a sample problem. The following is an overview of the methodology:

- Specify the tours of duty and the mealtime breaks that are permitted.
- Generate estimates of hour-by-hour requirements.
- Using the specifications made above, find a schedule using the fewest number of cars.
- Evaluate the schedule using information of the levels of police service that would result if the schedule were implemented.
- Based on this evaluation, revise the requirements and repeat as often as necessary.

ESTIMATING PATROL CAR REQUIREMENTS

Using almost any measure of effectiveness, patrol performance is improved by increasing the number of patrol cars on duty. Waiting times and response

times decrease while car availability and hours spent on preventive patrol increase. Most other indicators of the quality or quantity of patrol service exhibit improvement when the number of cars is increased.

The probability that all cars are busy is quantitatively related to the number of cars fielded. This relationship can be used to generate estimates of the number of cars required at each hour of the day to obtain desired police performance levels. Although it is difficult to exactly determine the number of cars needed to keep this probability below a given level, excellent estimates are provided by some mathematical approximations presented here. Schedules based on these approximations can be checked using the evaluation methods discussed later in this report. Therefore, we concentrate on the probability that all cars are busy as the main measure of performance in this report. However, our allocation and scheduling methodology can be applied to car requirements based on other service measures.

GENERATING AND EVALUATING SCHEDULES

Suppose that the number of patrol cars required in the field during each hour of the day has been specified. Suppose also that there is a set of acceptable tour start times given by the police department, as well as a set of acceptable mealtimes. We wish to find tour and mealtime assignments that meet the car requirements using the least number of cars. This can be accomplished with integer programming.

A schedule generated using integer programming is based partly on simple approximate models. Before trying any such schedule in the field, we would like to test it using mathematical models that represent more of the complexity of the real world. A mathematical model that can be used for this purpose is the time-dependent queueing model. With this model, waiting times, the probability that all cars are busy, and other statistics can be calculated. It requires as input the hourly call rate, the average time to service a job, and the number of cars on duty. The model uses a numerical solution procedure to calculate statistics for small time increments. Police department management can evaluate a schedule using these statistics and decide whether to change the car requirements.

SCHEDULING PRINCIPLES

In order to develop principles of good patrol car scheduling, a series of new schedules were generated and evaluated using the following characteristics: number of patrol cars used, number of calls for service, average number of patrol cars available for preventive patrol, convenience of the schedule, and patrol car hours available for all work.

The following general principles were derived from this analysis:

- Maximum benefits from schedule flexibility can be obtained using five different tour start times and assigning meals at any time.
- Schedules using three standard tours (starting at 0800, 1600, and 2400) and a fourth tour starting between 1800 and 2000 are nearly as effective as the best five-tour schedules.
- Flexibility in assigning mealbreaks results in a markedly improved pattern of availability.
- In almost all cases, preventive patrol hours can be doubled with only small percentage increases in the number of cars.
- The increased workload during the early hours of Saturday and Sunday can be handled by the addition of a few cars to the weekend tours covering these patrols.
- The principles summarized above can be generally applied because call rate patterns are similar throughout the city.

CONTENTS

PREFACE iii

SUMMARY v

I. INTRODUCTION 1

II. ESTIMATING PATROL CAR REQUIREMENTS 4

III. AN INTEGER LINEAR PROGRAM FOR GENERATING SCHEDULES 6

IV. EVALUATING SCHEDULES 9

V. A SAMPLE PROGRAM 14

VI. RESULTS: SOME SCHEDULING PRINCIPLES 23

APPENDIX A 43

APPENDIX B 45

I. INTRODUCTION

In recent years, a good deal of effort has been devoted to developing quantitative methods to help police departments allocate resources and schedule manpower. The work of Larson [13] and of Heller and Stenzel [7] are important examples. Surveys and evaluations of much of the pertinent literature are given by Chaiken [2] and by Gass et al. [5]. This report can be viewed as part of this general effort. It presents a methodology for creating new patrol car schedules that improve the correspondence between patrol car availability and demands for service.

The level of demand for police service varies considerably through the day. Recognizing this, police departments assign more patrol cars to duty during the busy hours. Figure 1 illustrates both of these points using data from one New York City police precinct. Scheduling decisions, however, have usually been more or less educated guesses. Improvements resulting from better schedules can be quite significant. In an example given below, a schedule derived using the methods presented in this report maintains performance standards with almost a 25 percent reduction in the number of cars fielded under the traditional schedule.

We view the patrol environment as a complicated multiple-server queueing system. Calls for service--either telephone calls to the radio dispatcher who sends the cars to the scene, or accidents, crimes, and other incidents encountered by the patrolling units--are assumed to occur randomly over time. These calls require random amounts of "service time" by one or more patrol cars. When not working on such "jobs" a patrol car is presumed to be either on preventive patrol or out of service for some reason. A variety of mathematical models of this environment can be formulated depending on the objectives of the analysis. Both queueing and optimization models have been used by other researchers [2]. Our objective is to use queueing theory to generate estimates of hourly car requirements needed as input to a scheduling model, and then to evaluate the resulting schedule with respect to various measures of patrol system performance. We discuss methods for generating estimates of patrol car requirements in Section II, an integer program

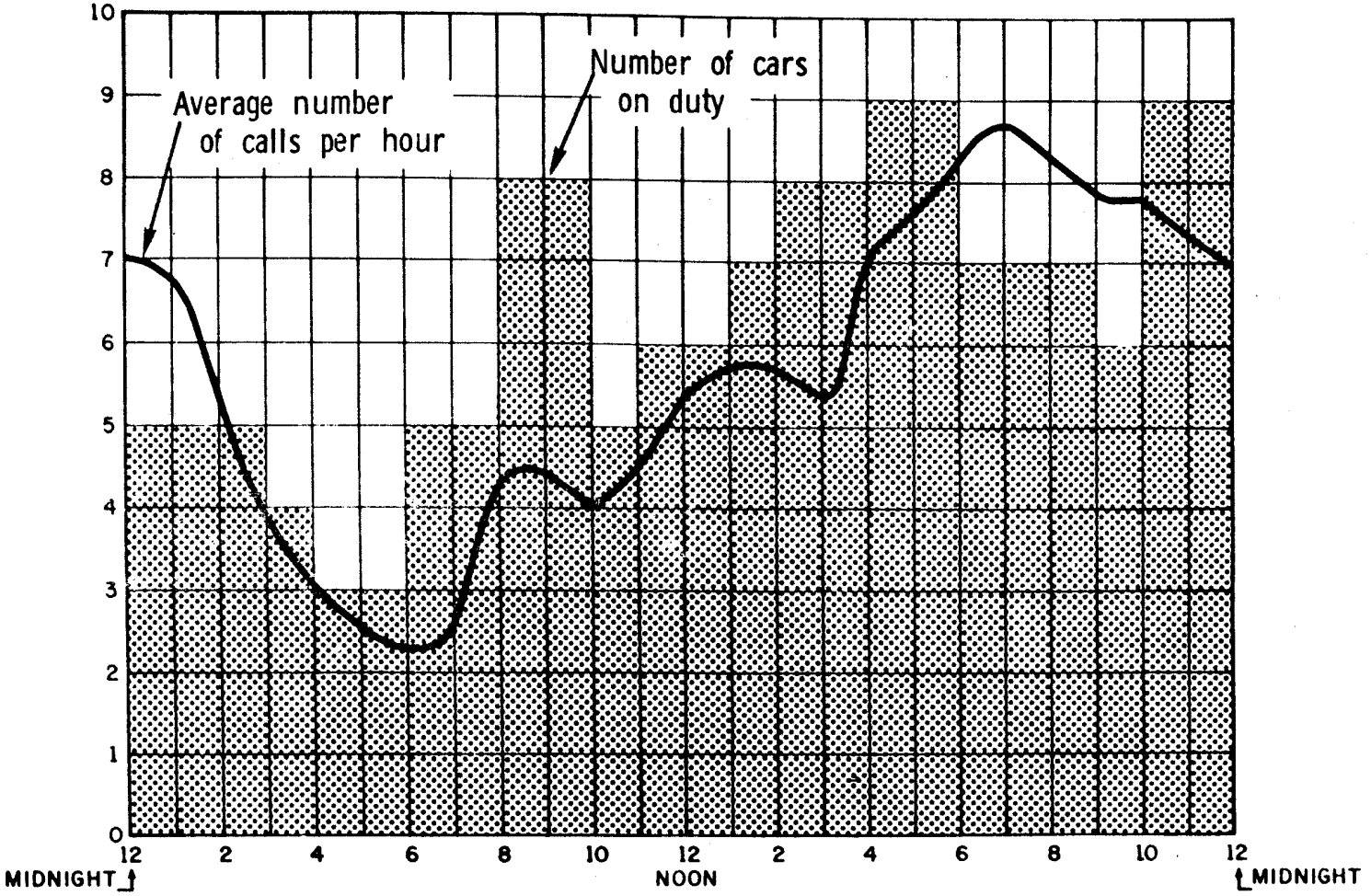

Fig. 1. Call rate and patrol car schedule as functions of the time of day in one New York City police precinct

for generating schedules given patrol car requirements in Section III, and a time-dependent queueing model for evaluating schedule performance in Section IV. Section V uses data from one of New York's police precincts to show how the method was applied to a sample problem.

The methodology that we have developed for creating and evaluating schedules is iterative. In broad outline, it consists of the following steps.

- *Specify the Policy Constraints on the Schedules.* That is, what tours of duty and mealtime breaks are permitted? (See Section V.)
- *Generate Estimates of the Number of Cars Needed in the Field by Hour.* This can be done using one of several mathematical models together with, and modified by, police judgment. (See Section II.)
- *Obtain an Optimal Schedule.* Using the specifications made above, an integer linear program is generated and solved that satisfies all of the constraints using the fewest number of cars. (See Section III.)
- *Evaluate the Schedule.* Using a time-dependent queueing model, detailed information is provided on the levels of police service that would be available over the day if the schedule were implemented. (See Section IV.)
- *Revise the Constraints.* Since approximate models were used to generate car requirements, the schedule generated may not actually meet all the performance levels desired. Moreover, there may be performance measures of interest other than those explicitly used in estimating patrol car requirements. This should be remembered when evaluating the resulting schedules. Based on this evaluation, the requirements and constraints of the integer program can be revised and the problem resolved. This process may be repeated as often as necessary. (See Section V.)

II. ESTIMATING PATROL CAR REQUIREMENTS

Using almost any measure of effectiveness, patrol performance is improved by increasing the number of patrol cars on duty. Queueing delays and response times decrease, while car availability and the hours spent on preventive patrol increase. Most other indicators of the quality or quantity of patrol service exhibit improvement where the number of cars is increased. In this section we show, as an example, how one specific measure of service performance is quantitatively related to the number of cars fielded, and how this relationship--really an approximation--can be used to generate estimates of the number of cars required at each hour of the day to obtain desired police performance levels. These requirement estimates can then be used as input to the integer programming model described in Section III. Similar methods can be used to derive car requirements based on other queueing-related service measures.

The relationship between the number of cars on duty during hour t and α, the probability that all the cars are busy, is complicated. The complications arise from the fact that we are dealing with the multiple-server queueing situation, in which the call rate, the number of cars on duty, and other factors are time-dependent. Suppose that the hourly call rates, λ_t, and service rates, μ_t, are known, and suppose also that α has been specified. We want to find r_t, the smallest number of cars to place on duty during hour t so that, given λ_t and μ_t, the probability that all r_t cars are busy at a random epoch during hour t is less than α.

This problem is difficult to solve, the primary difficulty being that the demand for service is not stationary in time, and so r_t depends on λ_t and also on λ_{t-1}, r_{t-1}, λ_{t-2}, r_{t-2}, ..., etc. A solution can be obtained using the time-dependent queueing model, which is discussed in Section IV, in an iterative trial and error fashion: Guess at the values of r_t; run the model; correct the values of r_t on the basis of the run; etc. In most cases, however, using the easier to calculate long-run (or stationary) state probability distribution, computed with parameters appropriate only to hour t, will provide excellent estimates of the required number of cars, r_t. There are two reasons why this is so. First, although queues building up in hour t may carry over into hour t + 1, if α is small the probability of a queue will also be small. Therefore, there will be little carry-over,

and hours t and t + 1 will be approximately independent. Second, although the call rate, λ_t, changes over time, the changes in successive hours are, in general, not large.

Let $a_j(t)$ denote the stationary probability that j jobs are in the system during hour t, given that there are n_t cars on duty, that the call rate is λ_t, and that the service rate is μ_t. Define $\rho_t = \lambda_t/\mu_t$. Then, using results for the M/M/n queueing model, $a_j(t)$ is given by:

$$a_j(t) = \begin{cases} \dfrac{\rho_t^{\,j}}{j!}\, a_0(t), & 1 \le j \le n_t \\ \dfrac{\rho_t^{\,j}\, a_0(t)}{n_t^{(j-n_t)}\, n_t!}, & j > n_t \end{cases}$$

and

$$a_0(t) = \left[\sum_{j=0}^{n_t-1} \frac{\rho_t^{\,j}}{j!} + \frac{\rho_t^{\,n_t}}{n_t!} \left(\frac{n_t \mu_t}{n_t \mu_t - \lambda_t} \right) \right]^{-1} .$$

Given a value of α, r_t is the smallest value of n_t such that

$$\sum_{j=0}^{n_t-1} a_j(t) > 1 - \alpha .$$

III. AN INTEGER LINEAR PROGRAM FOR GENERATING SCHEDULES

Suppose that the number of patrol cars required in the field during each hour of the day has been specified. (This may have been done using the formulae in Section II.) Suppose also that there is a set of feasible tour start times, S, given by the police department, all of which we assume start on the hour. Each tour of duty lasts eight hours. We also assume that every tour includes a one-hour mealtime, which also begins on the hour. There may be constraints on the earliest and latest hours of a tour that can be used for a meal break; let $e(t)$ and $\ell(t)$, respectively, denote these values for a tour starting at hour t. Then the earliest possible mealtime for cars working tour t starts at hour $t + e(t)$, and the latest possible mealtime begins at $t + \ell(t)$. (These assumptions are made for clarity of exposition only and are not intrinsic to the model. The model can easily handle tours starting on the half hour, quarter hour, etc., as well as tours and mealtimes of any length.)

Consider a 24-hour problem, and let r_t denote the (integral) number of cars required during hour t ($t = 0,1,\ldots,23$; hour 24 = hour 0), where hour t runs from t to $t + 1$ clock hours. The decision variables of the programming problem are:

n_i = the number of cars assigned to work the tour starting at hour i, $i \in S$.

m_{ij} = the number of cars working tour i that are assigned to mealtime at hour j, where $i+e(i) \leq j \leq i+\ell(i)$ and $i \in S$.

We wish to find tour assignments $\{n_i\}$ and mealtime assignments $\{m_{ij}\}$ that meet the car requirements using the least number of cars. An integer linear program that accomplishes this objective is:

$$\text{minimize} \quad \sum_{i \in S} n_i$$

subject to

$$\sum_{j=i+e(i)}^{i+\ell(i)} m_{ij} = n_i, \qquad i \in S$$

$$\sum_{\substack{i \in S \\ t-7 \le i \le t}} n_i - m_{it} \ge r_t, \qquad t = 0, 23 \qquad \text{(A)}$$

$$n_i \ge 0, \; m_{ij} \ge 0, \text{ and integer.}$$

The value of the objective function is the total number of cars used over the day. The first set of constraints assures that every car is assigned a mealtime, and the second set of constraints assures that the actual cars on duty (number of cars assigned minus number of cars on meals) meets each of the hourly requirements.

The resulting integer linear programs can be quite large for reasonable real-world situations. With only three tour start times and four possible mealtimes, there are 15 variables and 27 constraints. This is modest but not trivial for an integer program. With 24 tours, and mealtimes allowed at any hour of a tour, there are 216 variables and 48 constraints. We have also formulated and solved problems for an entire week involving as many as 1344 variables and 336 constraints.

Fortunately, the constraint matrix has a special structure that permits the problem to be solved as a mixed integer program in which only a small number of the variables need be "forced" to be integers. The remaining variables are automatically integral in any optimal solution. This permits use of a standard mixed integer programming code to solve even the largest of our problems in reasonable times. The key idea is contained in the following:

Theorem: Suppose that n_i, the number of cars assigned to tour i, is limited to integer values. Then m_{ij}, the mealtime assignment variables, are automatically integral in any basic solution of (A).

Proof: When the n_i take on fixed values, say n_i^*, the constraints of (A) become

$$\sum_{j=i+e(i)}^{i+\ell(i)} m_{ij} = n_i^*, \quad i \in S$$

$$\sum_{i \in S} m_{ij} \leq -r_t + \sum_{\substack{t-7 \leq i \leq t \\ i \in S}} n_i^*, \qquad t = 0, 23 \tag{B}$$

The only coefficients of (B) are +1 and 0, and each of the columns of the associated matrix contains at most two nonzero elements. The matrix, therefore, has a network-like structure that is well known to be totally unimodular. Hence, all of the extreme points of (B) are integer regardless of the values of r_t and n_i^* ([4], p. 70; [8], p. 126).

To solve the smaller problems, the mixed integer program (MIP) subroutines of MPSX [14] can be used in a straightforward fashion. To solve the larger problems we have modified the branching rules in MIP to take advantage of the problem structure. See Appendix A for a description of these modifications. An overview of the MIP-MPSX algorithm is given by Geoffrion and Marsten [6].

IV. EVALUATING SCHEDULES

The actual environment in which a schedule may be implemented is complex and random. Nevertheless, a schedule generated using the approach described above is derived from a deterministic integer linear programming model, the inputs to which are determined partly by simple approximate models and partly by police judgments. Before trying any such schedule in the field, we would like to test it using mathematical models that represent more of the complexity of the real world. Two types of models can be used for this purpose. The first is a simulation model of police patrol operations, such as the one described in [10]. A simulation, however, is a relatively expensive and cumbersome tool that requires the collection of a considerable amount of input data to make it work.

The second type of model that can be used for evaluation is a time-dependent M/M/n queueing model. It is not difficult to develop the set of differential equations that describes the system dynamics of the M/M/n queueing system with time-dependent parameters. It is, however, extremely difficult to obtain an analytic solution to the set of equations. We therefore propose numerical integration of the differential equations to obtain such characteristics of the patrol system as the probability distribution of the number of busy patrol cars and the number of calls queued. Our approach is motivated by a paper by Bernard O. Koopman [11] in which the efficiency and usefulness of this type of model is illustrated by applying it to the study of air traffic control problems. Koopman also discusses the advantages of this approach relative to the use of simulation.

Our time-dependent queueing model represents a single police precinct. As the different tours and mealtimes commence, the number of police cars available for patrolling and for servicing calls will vary. The rate at which calls for service are received also varies during the course of the day, with a peak occurring in the late evening hours and a lull in the morning. (See Fig. 1.) From historical data we can predict the average number of calls for service and the distribution of the number of calls during each hour of the day. Our data also permit identification of daily and seasonal patterns in the call rate. We have determined from these

call histories that the arrival of calls for service in any given hour can be well represented as a Poisson process. That is, the probability that a given number of calls will occur during a given hour in a given day is specified by a Poisson distribution with a mean characteristic of that time period.

The model assumes that each call for service is handled by a single patrol car. In practice it sometimes happens that two or more cars are necessary to service a call. Nevertheless, a comparison of results from a stationary M/M/n queueing model (which also incorporates the one car per call assumption) to results from the simulation model (which more closely imitates reality and uses as many cars as is appropriate for the call) indicates that such complications may be neglected without seriously altering the resulting queueing probabilities [9].

While the types of calls (crimes in progress, past crimes, emergencies, accidents, etc.) may vary during the day, we have found that the average service time remains fairly constant. Actual data show that the service times for calls do not have an exponential distribution as assumed by the model, but again a comparison of the M/M/n queueing model to the simulation model, which used empirical--and hence nonexponential--service times, shows good agreement for the prediction of average performance.

One limitation of the time-dependent queueing model is that there is no priority structure in the dispatching of calls. As a result, we can only examine overall call delays. Modeling of priority calls is possible, but it would significantly increase the complexity of the computations. It was not undertaken since detailed analysis of delays by call priority can be performed with either the simulation or a stationary M/M/n queueing model. Calls in this time-dependent model are served, therefore, on a first-come-first-served basis. If a patrol car is free when a call arrives, it is dispatched immediately and the call remains in the system only for the length of its service time. (The response time of the patrol car is not explicitly modeled. It would be very difficult to do this in an accurate way without destroying the

Markovian nature of the system, which is essential to our method of solution. Since travel times are short compared to the time spent at the scene of the incident, this approximation is not critical.) During the service interval the patrol car is unavailable for further assignments. If a call arrives when all patrol cars are busy, it waits in queue until all preceding calls have been dispatched. It is then dispatched as soon as the next car becomes available.

The model we have just described is an M/M/n queueing system with time-dependent parameters. In studying this system, we focus attention on the random variable X(t), the number of calls in the system at time t, including those being served by patrol cars and those in the queue. A great deal of information about the performance of the system can be obtained from the Markovian transition probability function $p_{ij}(t_o,t)$, defined as follows:

$$p_{ij}(t_o,t) = P[X(t)=j \,|\, X(t_o)=i],\ t_o \geq 0,\ t>t_o,\quad i,j = 0, 1, \ldots$$

For example, let n(t) denote the number of patrol cars on duty at time t. Then if X(t) is less than n(t), X(t) represents the number of busy cars and n(t) - X(t) represents the number of cars on patrol. If X(t) is greater than or equal to n(t), all cars are busy and X(t) - n(t) represents the number of calls waiting in the dispatching queue. Suppose that at some time we know that there are i calls in the system--that is, $X(t_o) = i$. Given this information about the state of the system at time t_o, we can calculate the following system performance characteristics for any future time t:

The probability that there is at least one call in queue:

$$P[X(t)\geq n(t)+1] = \sum_{j=n(t)+1}^{\infty} p_{ij}(t_o,t) \qquad (1)$$

The probability that all cars are busy:

$$P[X(t)>n(t)] = \sum_{j=n(t)}^{\infty} p_{ij}(t_o,t) \qquad (2)$$

The expected number of calls in queue:

$$EQ = \sum_{j=n(t)+1}^{\infty} [j-n(t)]p_{ij}(t_o,t) \qquad (3)$$

The expected number of cars available for patrol:

$$EA = \sum_{j=0}^{n(t)-1} [n(t) - j]p_{ij}(t_o,t). \qquad (4)$$

Before considering how to determine the transition probabilities, we introduce some additional notation:

$\lambda(t)$= the call rate at time t; i.e., the expected number of calls per hour being received at time t, which is the mean of the Poisson process generating the calls. Here we refer to a specific epoch t and its instantaneous call rate $\lambda(t)$. Earlier we used λ_t to refer to the average number of calls during the hour t to t + 1.

μ = the service rate; $1/\mu$ =ES, the expected service time per call. As verified by actual data, we assume in this analysis that μ does not change through time. (Relaxation of this assumption would not appreciably complicate the analysis.)

The transition probabilities satisfy the following system of differential-difference equations. For $t>t_o$,

$$p_{io}'(t_o,t)=-\lambda(t)p_{io}(t_o,t)+\mu p_{i1}(t_o,t),$$

$$p_{ij}'(t_o,t)=\lambda(t)p_{ij-1}(t_o,t)-[\lambda(t)+j\mu]p_{ij}(t_o,t)+\mu(j+1)p_{ij+1}(t_o,t), \quad 1\leq j<n(t), \qquad (5)$$

$$p_{ij}'(t_o,t)=\lambda(t)p_{ij-1}(t_o,t)-[\lambda(t)+n(t)\mu]p_{ij}(t_o,t)+n(t)\mu p_{ij+1}(t_o,t), \quad j\geq n(t).$$

These equations cannot be solved analytically for $p_{ij}(t_o,t)$, except for the simplest of functions $\lambda(t)$ and $n(t)$. However, we can solve (integrate) them numerically. Since $\lambda(t)$ and $n(t)$ are periodic functions (repeating themselves every 24 or 168 hours depending on the application) there is a periodic solution that is independent of the initial state i. We wish to find this periodic solution. We denote it by $p_j(t)$, the "long run" probability that $x(t)=j$. ($p_{ij}(t_o,t)$ approaches $p_j(t)$ for large t. [7])

There are an infinite number of equations in (5). In order to solve them numerically we limit ourselves to a finite system of equations that approximates (5) by assuming--not unrealistically--that there is a maximum possible number of calls, m, that can be in the system at one time. In some applications the value of m is dictated by the limitations of the dispatching system. Where such physical constraints do not exist, m is chosen so that the probability of having m or more calls in the system is very small. Hence, we replace (5) by

$$
\begin{aligned}
p_o'(t) &= -\lambda(t)p_o(t)+\mu p_1(t) \\
p_j'(t) &= \lambda(t)p_{j-1}(t)-[\lambda(t)+j\mu]p_j(t)+(j+1)\mu p_{j+1}(t), \quad 1\le j<n(t) \qquad (6) \\
p_j'(t) &= \lambda(t)p_{j-1}(t)-[\lambda(t)+n(t)\mu]p_j(t)+n(t)\mu p_{j+1}(t), \quad n(t)\le j<m \\
p_m'(t) &= \lambda(t)p_{m-1}(t)-n(t)\mu p_m(t).
\end{aligned}
$$

A discussion of the numerical methods used to solve this set of m+1 differential equations is given in Appendix B.

V. A SAMPLE PROBLEM

We illustrate the scheduling methodology described above with a sample problem based on data from a police precinct in New York City. The precinct used in this example cannot be called typical--there is no such thing. It does, however, have characteristics in the middle range of precincts on several measures: physical area, total demand for police service, crime rate, number of cars fielded, etc.

Our data for the hourly call rates are derived from job records collected during one week in August 1972 by the computerized dispatching system used by the New York City Police Department. Based on empirical data, we took the average service time to be 30 minutes (μ = 2 calls per hour).

Table 1 contains λ_t, the call rate, $\rho_t = \lambda_t/2$, the average number of busy cars, and r_t, the number of cars required so that the system is unclogged at least 90 percent of the time. That is, r_t is the smallest number of cars needed during hour t so that the probability of at least one car being available to respond to a call is at least 0.9. The values of r_t were estimated using the stationary M/M/n queueing model, as discussed in Section II.

Figure 2 shows the schedule that was actually in use in the precinct during the period in August 1972 from which our data come. It uses 24 cars over the three eight-hour tours of a day. Because of low car availability during the early morning hours, the schedule produces periods in which there is a very high probability that no cars will be available to answer a call for service. The line on the graph that shows the probability that there are no cars on patrol was obtained from the time-dependent queueing model.

In order to obtain schedules with better performance characteristics, we set up and solved several integer linear programs. A description and evaluation of some of the resulting schedules follows.

INTEGER LINEAR PROGRAM 1 (STANDARD TOURS WITH STANDARD MEALTIMES)

In this case, we restricted ourselves to the tour start times and mealtimes generally used in the New York City Police Department. There are three permitted tour start times--0800, 1600, and 2400 (or 0000) hours--and mealtimes can be taken between the second through the fifth hours of the

Table 1

HOURLY CALL RATES, EXPECTED NUMBER OF BUSY CARS, AND NUMBER OF CARS REQUIRED FOR SAMPLE PROBLEMS

t	λ_t	ρ_t $(\lambda_t/2)$	r_t
0	9.8	4.9	9
1	9.6	4.8	9
2	8.7	4.4	8
3	7.4	3.7	8
4	6.7	3.4	7
5	5.3	2.7	6
6	4.1	2.1	5
7	3.3	1.7	4
8	2.5	1.3	4
9	2.5	1.3	4
10	2.9	1.5	4
11	3.8	1.9	5
12	4.3	2.2	5
13	5.0	2.5	6
14	5.9	3.0	6
15	6.6	3.3	7
16	7.8	3.9	8
17	8.6	4.3	8
18	9.4	4.7	9
19	9.8	4.9	9
20	10.2	5.1	9
21	10.4	5.2	9
22	10.2	5.1	9
23	10.0	5.0	9

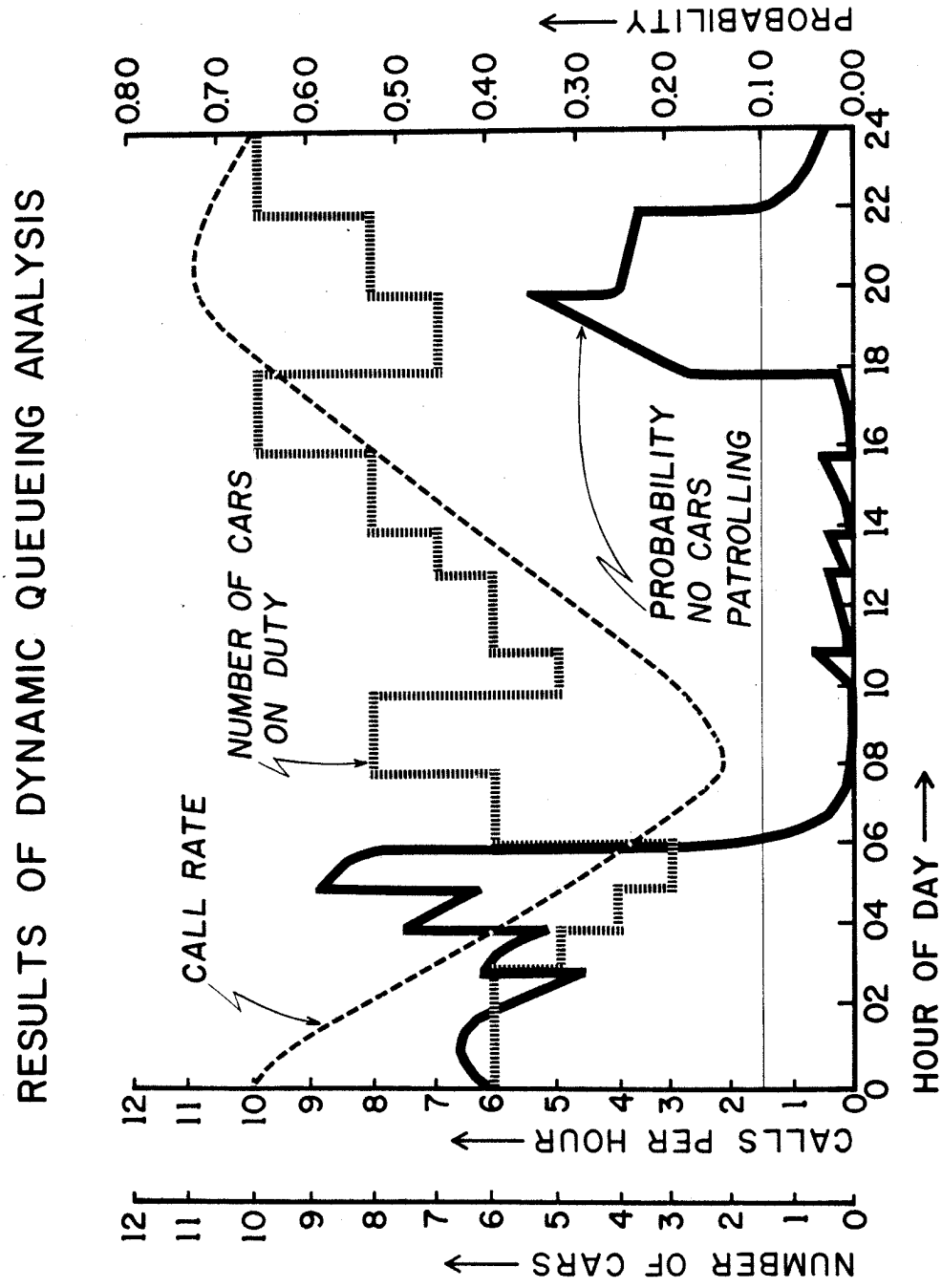

Fig. 2. An actual NYPD schedule using 24 cars

tour. There are 15 variables and 27 constraints in the linear program. Using the car requirements, r_t, shown in Table 1, we obtain the following optimal integer solution, which uses 29 cars:

(i) Tour starting at 2400 hours

cars assigned = 10

mealtimes assigned:
1 cars at 0200
2 " " 0300
3 " " 0400
4 " " 0500

(ii) Tour starting at 0800 hours

cars assigned = 7

mealtimes assigned:
2 cars at 1000
2 " " 1100
2 " " 1200
1 " " 1300

(iii) Tour starting at 1600 hours

cars assigned = 12

mealtimes assigned:
3 cars at 1800
3 " " 1900
3 " " 2000
3 " " 2100

The characteristics of this schedule are illustrated in Fig. 3. Here again, the probability that there are no cars on patrol was computed using the time-dependent queueing model.

INTEGER LINEAR PROGRAM 2 (ALL POSSIBLE TOUR START TIMES AND MEALTIMES)

This program allows a tour to start at the beginning of any hour of the day, and a unit's mealtime can be taken during any hour of the tour. The solution to this problem provides the smallest possible number of cars that could be used to meet the specified requirements. Any additional restrictions, such as prohibited hours for starting mealtimes or tours, will produce a requirement for at least as many cars as the solution to this program.

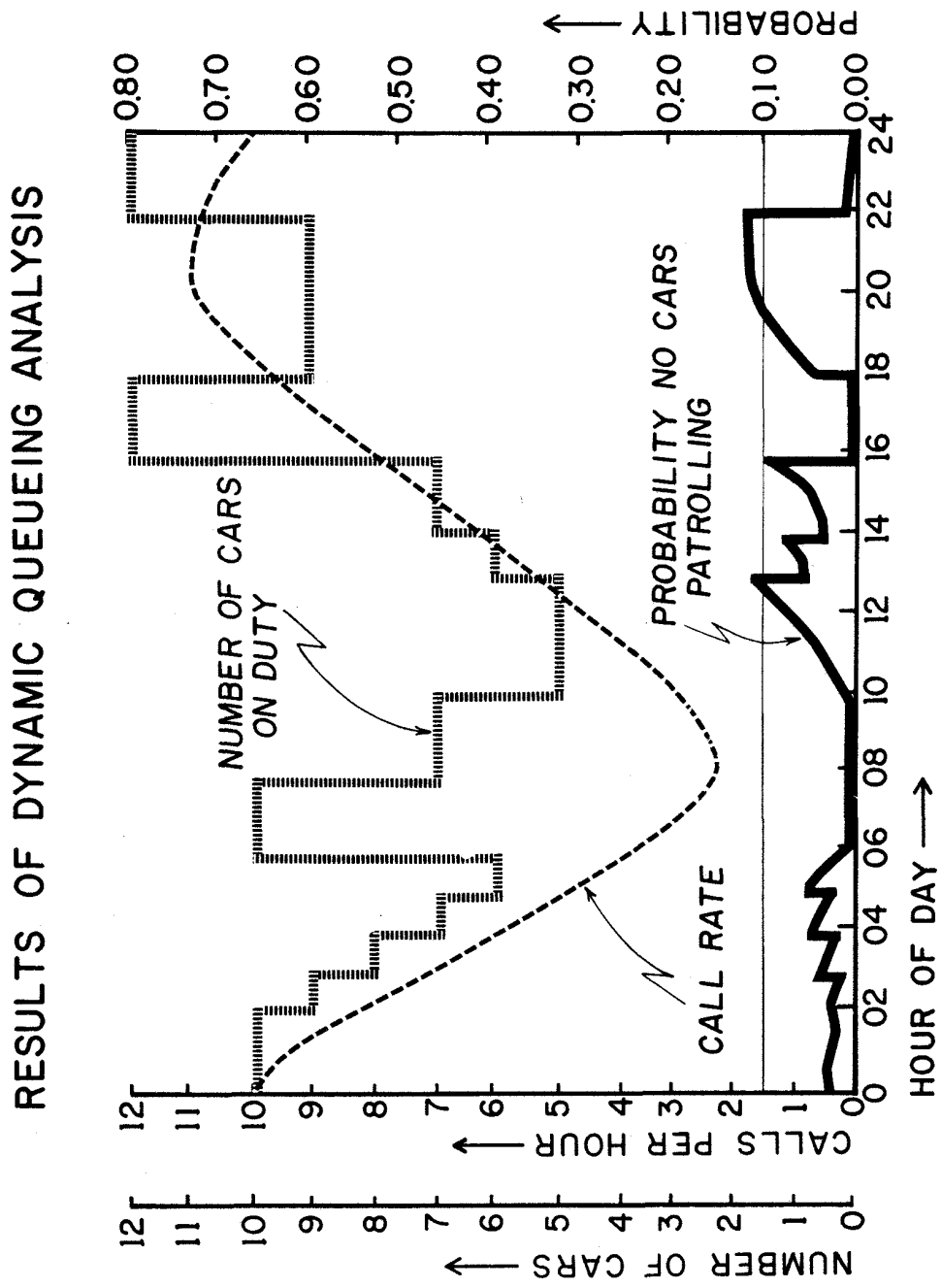

Fig. 3. A computer-generated 29-car schedule with three tours and $\alpha = 0.1$.

An optimal integer solution was obtained that requires 24 cars over the day, but uses 13 different tour start times--an administrative nightmare to implement. This schedule is not illustrated here.

This result led us to ask if it was possible to find a less difficult schedule, but one that would still require only 24 cars. The type of schedule we wanted would have only a small number of tour start times, avoid undesirable start times (e.g., 3:00 a.m.), and have mealtimes scheduled during reasonable hours of the day. By running programs with different numbers of tours and different allowable mealtimes, relying on the results of the previous analysis for insights on what might work well, we obtained the following schedule:

INTEGER LINEAR PROGRAM 3 (FIVE TOURS, ALL POSSIBLE MEALTIMES)

This program has five tour start times--the three current times, 0800, 1600, and 2400, and two additional times, 1200 and 2000. Mealtimes are allowed at any hour during a tour. The optimal integral solution calls for a total of 24 cars (the minimum possible) and the solution is:

(i) Tour starting at 0800 hours

cars assigned = 5

mealtimes assigned: 1 cars at 0800
1 " " 0900
1 " " 1000
1 " " 1200
1 " " 1400

(ii) Tour starting at 1200 hours

cars assigned = 2

mealtimes assigned: 1 cars at 1200
1 " " 1300

(iii) Tour starting at 1600 hours

cars assigned = 8

mealtimes assigned: 2 cars at 1600
2 " " 1700
1 " " 1800
1 " " 1900
1 " " 2200
1 " " 2300

(iv) Tour starting at 2000 hours

cars assigned = 2

mealtimes assigned: 1 cars at 2000
1 " " 2100

(v) Tour starting at 2400 hours

cars assigned = 7

mealtimes assigned: 1 cars at 0300
1 " " 0500
2 " " 0600
3 " " 0700

This schedule is illustrated in Fig. 4. Comparing these results to the current situation shown in Fig. 2, we see that although both schedules field the same number of patrol cars over a day, the resulting performance characteristics are considerably different. For example, under the current schedule, almost 60 percent of the incoming calls during some hours have to wait in queue before a car is dispatched. Under the schedule resulting from Linear Program 3, the percentage of calls delayed in queue never exceeds 12 percent.

The above are only a few examples from an extensive series of integer linear programs that were solved to test and develop new schedules for the New York City Police Department. The resulting schedules that appeared capable of being implemented were subjected to analysis using the time-dependent queueing model. In addition to the one-day schedules illustrated above, we solved scheduling problems for entire weeks with data from different seasons of the year and from different precincts. We used the techniques presented in this paper to find answers to such questions as:

- What improvements in performance would result if patrol cars could be assigned to start their tours of duty at any hour of the day and take their mealtimes at any hour during the tour?
- What is the best schedule that uses only four tour start times? How much worse is the performance of such a schedule than the performance of the best five-tour schedule? How much better is this schedule than the standard three-tour schedule?
- Since the pattern of calls on weekends is different from weekdays, should the weekend schedules be different?

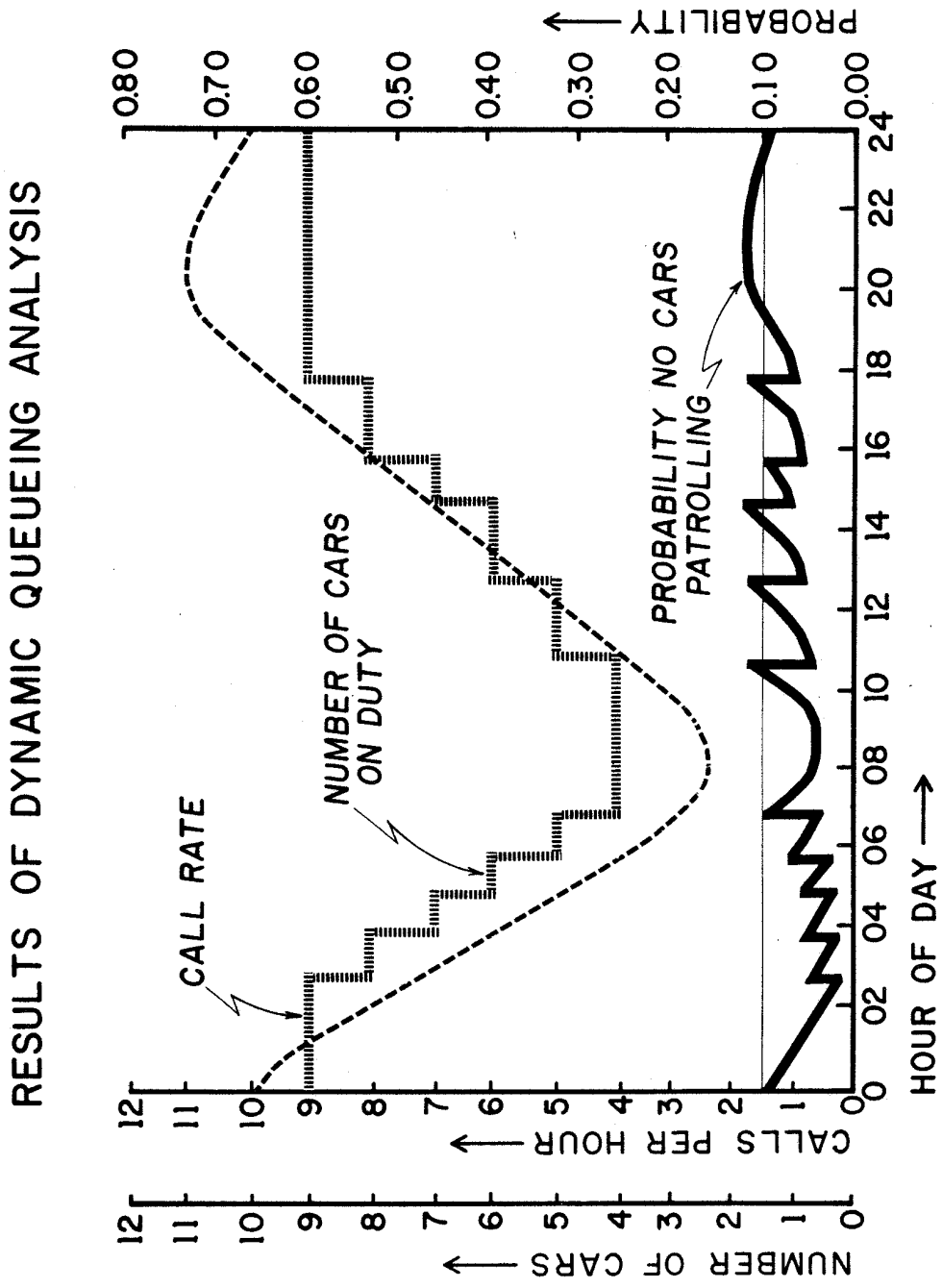

Fig. 4. A computer-generated 24-car schedule with five tours and $\alpha = 0.1$.

- Are the patterns of calls for service in different areas similar enough so that principles developed from studying a few precincts can be applied generally?
- How does the number of patrol cars assigned vary with the desired service levels?

VI. RESULTS: SOME SCHEDULING PRINCIPLES

In order to generate improved patrol car schedules and to develop general principles of good scheduling, we undertook a series of analyses with the linear programming and queueing models. In this section of the report we discuss the results of this work. The many new schedules that were generated during this analysis are evaluated and compared using the following characteristics:

- The number of patrol cars used. This is the measure of resource input. We denote it by N.
- The service level. This is the probability that all patrol cars are busy simultaneously. We denote it by α.
- Work time. This is the time available to each fielded car to answer calls for service and undertake preventive patrol --after accounting for meals, vehicle down-time, administrative duties, etc. We denote it by h.
- The average number of patrol car hours available for (preventive patrol. This is the difference between total work time (Nh) and average total patrol car hours required to service all jobs. For a 24-hour day

$$A = Nh - 1/\mu \int_0^{24} \lambda(t)\, dt.$$

- The convenience or attractiveness of the schedule to the patrolmen and to management. For example, a schedule with seven different tour starting times might rate highly on the characteristics cited above, and yet be unusable because of its unattractiveness to the men and its unwieldiness to manage and implement.

Before getting into the details, we present an overview of the questions raised and the answers provided by the analysis.

- Schedules Using Minimum Patrol Resources

  The question: What kind of schedules would be used and what improvements would result if patrol cars could be assigned to

start their tours of duty at any hour of day and take their meal breaks at any hour during the tour?

The answer: The maximum benefits can be obtained by using five different tour start times and by assigning patrol cars to take meals at any hour of the tour (including the first and last hours). These schedules can achieve the same service level α as the standard three-tour schedule with standard meal breaks, but with about 20% fewer cars. If the same number of cars are used as in a standard three-tour schedule, these new schedules can eliminate undesirable variations in the service level.

- o Schedules Using Four Tours

The question: What is the best schedule that uses only four tour start times? How much worse is such a schedule than the best five-tour schedule? How much better is this schedule than the standard three-tour schedule?

The answers: The best four-tour schedules have tours starting at the three standard times of 0800, 1600, and 2400, and a fourth tour starting between 1800 and 2000. Such four-tour schedules are nearly as effective as the best five-tour schedules. They have marked advantages over the standard three-tour schedule: They provide more patrol during the evening hours of high crime, and they result in a very much lower probability that no cars are available during the early morning hours when the call rate is still relatively high. (These are the hours when the standard three-tour schedule's performance is poorest.) These observations are valid regardless of the mealtime schedules in force, as long as the same mealtime scheduling rules are in force for all tour arrangements.

- o Scheduled Mealtimes

The question: What performance benefits result from increasing the range of hours during which meal breaks are taken? (In New York City, unlike some other municipalities, each policeman has a scheduled meal hour during his tour of duty.)

The answer: Increased flexibility, even to the extreme of assigning meal breaks in the initial or final hour of the tour (thus permitting the unit to be on duty for seven rather than eight hours), markedly improves the pattern of availability. The magnitude of this effect is equivalent to that of several additional cars.

- o How Patrol Car Requirements Vary with Desired Performance Level

The questions: How many additional patrol units are needed to, say, double the number of hours of preventive patrol, or, say, halve the service level, α?

The answers: Since the total workload of answering calls is fixed, any additional patrol car hours beyond those necessary to handle calls are additional hours available for preventive patrol. So, in almost all cases the number of preventive patrol

hours can be doubled with only small percentage increases in the number of cars. On the other hand, halving the service level will usually require very significant increases in patrol cars. Graphs of both available patrol hours and service level as a function of the number of cars on patrol are given in Fig. 5 for a typical precinct.

- o Weekend Schedules

  Question: Since call rates on weekends are different from weekdays, how different should weekend schedules be?

  Answer: Increased workloads encountered in the early hours of Saturday and Sunday mornings can be handled by adding a few cars to the tours covering these periods.

- o Similarity of Scheduling Principles for Different Precincts

  Question: Do the patterns of demand for patrol car services differ so much from precinct to precinct that general principles and extrapolations cannot be made from a few "typical" precincts?

  Answer: From the analysis of demand patterns in six precincts chosen to be representative in New York City, we conclude that the principles summarized above can be generally applied and that the single most important characteristic in scheduling is the overall call rate.

To answer the above questions we generated two types of schedules:

(1) Schedules for a "typical" or average day. The call rates used in generating these schedules are averages by hour over all days of the week. The linear programming model consists of 24 one-hour time periods. We used this model for most of our analyses since it can provide answers to many of the questions posed, and yet is much more economical to run the larger model described below.

(2) Schedules for a "typical" or average week. The call rates are the averages for each hour (by day of week) during the week. The linear programming model consists of 168 one-hour time periods. This model is quite large and more expensive to solve. Its use was therefore restricted to questions about weekday vs. weekend scheduling and to checking observations and principles that seemed to be valid on the basis of our analysis of an average day with the smaller 24-hour linear program.

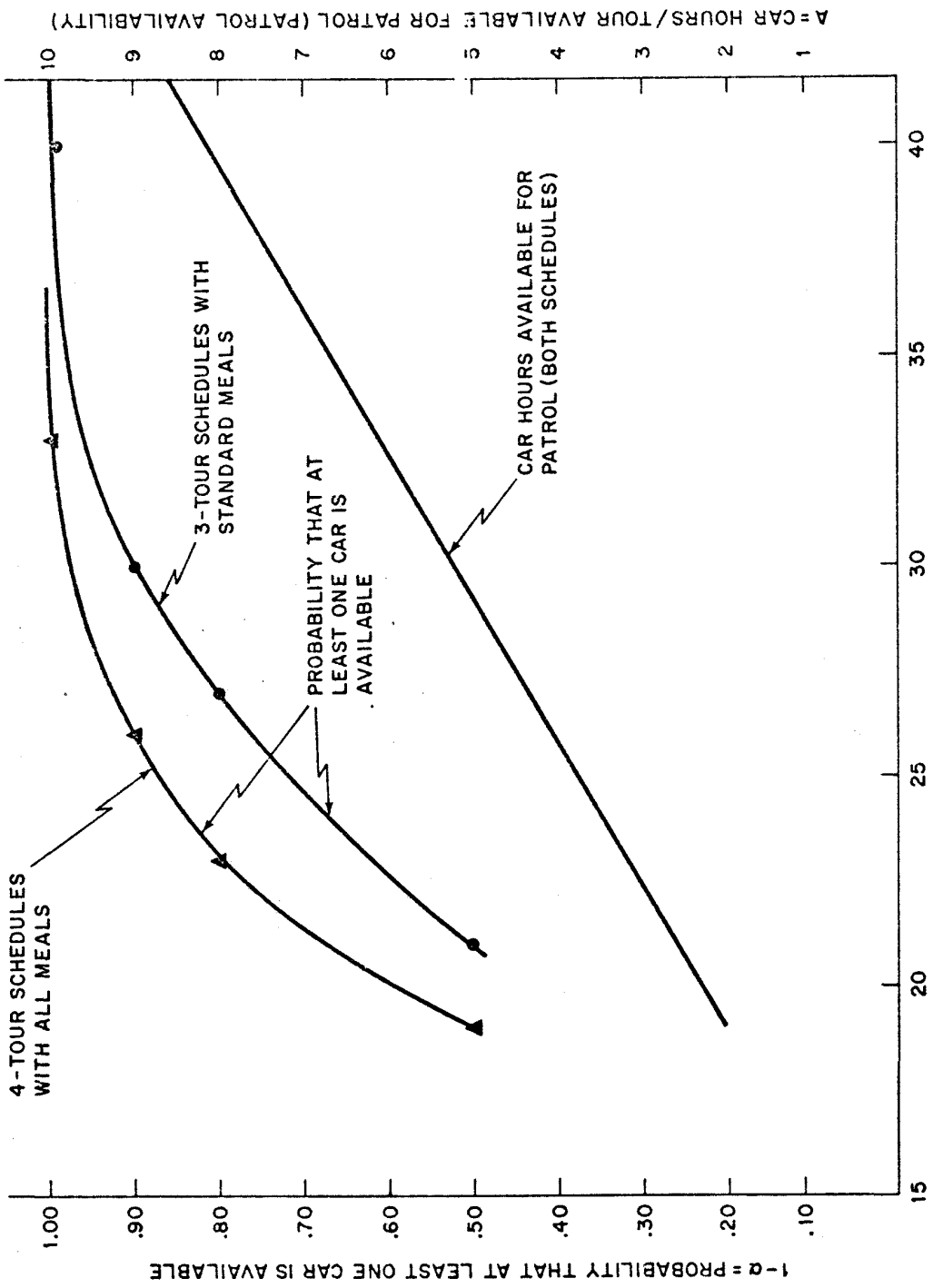

Fig. 5. Patrol car performance measures as a function of the number of cars assigned

In most of our analyses we used the average number of calls for service received per hour through 911 in the summer of 1972 in the 71st Precinct. Because we did not include "pick up jobs" or other incidents not initiated by a call through 911, workload (or demand) is understated. On the other hand, there is some reason to believe that the job service times, also estimated from data for 911 calls, are overstated. The key input parameter in the models is the product of the call rate and the average service time, and so the two effects tend to cancel each other, although not completely. Moreover, the principles we drew from the analysis do not depend in a crucial way on precise knowledge of these characteristics. It is the general level of calls and more importantly their pattern through the day that are most significant. So we believe that the principles are valid, even though the data may be somewhat imprecise, and that the effort required to obtain more exact data would not have been warranted. Even though the Department has a computer-aided dispatch system, only data for 911 calls are recorded. The collection of data for all calls would have been a difficult manual task.

MINIMUM CAR SCHEDULES

According to logs from the NYPD Communications Division, the average number of cars fielded per day in the 71st Precinct during that period was 24: six cars on the midnight to 8 a.m. tour, eight cars on the 8 a.m. to 4 p.m. tour, and 10 cars on the 4 p.m. to midnight tour. Figure 2 shows the time-dependent queueing model results for this schedule. The probability that all cars are busy peaks strongly at several points, and is well above 0.1. We solved a linear programming model that permitted tours to start at any hour of the day and meal breaks to be assigned during any hour of the tour, and set the service level $\alpha = 0.1$. The resulting schedule used a total of 24 cars assigned to five tours, beginning at 0800, 1200, 1600, 2000, and 2400 hours. It was somewhat surprising that only 5 of the 24 possible tour start times were used. This result seems to hold promise that effective yet simple schedules can be generated.

However, the complete flexibility of mealtime assignments was used in the linear program solution and some cars were assigned to take meal

breaks during the first and last hours of their tours. This is a less pleasing result, since such meal breaks may be impossible to implement. Figure 3 shows the time-dependent queueing model results for this schedule. The peaks observed in Fig. 2 in the probability that all cars are busy have been lowered, and the function lies at or below 0.1 throughout the day--a considerable improvement over the other 24-car schedules.

In order to determine how much of this improvement was due to the new schedule of meal breaks, we also solved a linear program identical to the one above in all respects except that meal breaks were restricted to the usual hours--from the third to the sixth hours of tour. This schedule required three more cars--a total of 27. In subsequent sections we discuss the issue of mealtime scheduling further.

BEST FOUR TOUR SCHEDULES

In the past, New York City has experimented with four tour schedules. The "best" schedule we have developed uses only five tours. Moreover, this schedule includes the three standard tours. Therefore, we ran several series of linear programs to determine the best possible four tour schedules. In the first series all meal break times were permitted. In the second series only the (currently used) four middle hours of the tour were permitted. In the third series meal breaks were permitted during the middle six hours of the tour, and in a fourth series meal breaks were permitted in all hours except the second and seventh. The key results are summarized in Table 2, which shows that a four-tour schedule using the standard three tours starting at 0800, 1600, and 2400 hours, and an additional tour starting at 2000 hours, is best as it uses the fewest cars under all mealtime regimes. The results in the table also show that a fourth tour starting at other hours between 1800 and 2200 hours may be equally good, depending on the meal break regime used. Other results not presented here showed that no other four-tour schedules are superior to those that use the three standard tours plus an additional tour beginning in the late afternoon or early evening.

Table 2

NUMBER OF CARS REQUIRED ON SOME FOUR-TOUR SCHEDULES
(α = 0.1 AND THREE TOURS STARTING A 0800, 1600 AND 2400 HOURS)

Fourth Tour Starting Time	Meals Permitted During Tour Hours			
	1 - 8	3 - 6	2 - 7	1, 3 - 6, 8
none[a]	26	29	28	28
1800	26	28	27	27
1900	26	28	27	26
2000	25	28	27	26
2100	25	29	27	27
2200	25	29	26	Not Run

[a]Three-tour schedule.

MEAL BREAK SCHEDULING

The data in Table 2 show that significant gains result if meal breaks can be spread out more evenly during the tour. Evidence of this can be seen by examining the top row of the table which gives, for different meal break regimes, the number of cars required when the standard three tours are used. Where meals are taken only during the middle four hours of the tour, 29 cars are needed, while only 26 cars are needed if meal breaks may be taken during any hour of the tour. Under the other two regimes, 28 cars are needed. Similar evidence of the impact of meal scheduling can be seen in the row corresponding to schedules with a fourth tour starting at 2000 hours. With the standard meal schedule 28 cars are needed. When all hours are usable 25 cars are needed, while 27 cars are needed when meals are permitted in hours 2 through 7 and 26 cars are needed when meals are permitted in all hours except 2 and 7. Other results not included in the table show that permitting meal breaks in hours 2-8 of the tour yields schedules using the same number of cars as when meals are limited to the hours 2-7 of the tour.

It is conceivable that the assignment of meal breaks during the first or last hour of the tour--an option that at first seems bizarre--might be

an attractive option. Suppose that men assigned to "take meals in the first hour of the tour" were permitted to arrive an hour later and work for seven consecutive hours. They would have a seven-hour work day. Except under the most extraordinary of circumstances, the nature of motorized patrol permits the taking of light snacks or sandwiches without interrupting patrol. In fact, this is common practice at the present time. The same seven-hour work day option could apply to those men whose meals are scheduled for the last hour of the tour.

We note that Table 2 shows that the most effective options are those that combine increased flexibility in meal scheduling *and* a fourth tour.

PATROL CAR REQUIREMENTS AND DESIRED PERFORMANCE LEVELS

Patrol performance increases with more cars. Here we examine the nature of this relationship for two key performance measures:

- The amount of preventive patrol;
- The probability that all patrol cars are busy.

Preventive Patrol

The number of patrol cars that must be added to yield a specified increase in car hours available for preventive patrol can be calculated as follows. We consider a single tour (or other time period of interest) and assume that the same number of cars are fielded throughout the tour. The "tour" is t hours long and each fielded car yields h hours of "work time," ($h \leq t$). Suppose that calls for service occur at a rate λ per hour and $1/\mu$ is the average service time in patrol car hours per call. During a tour of length t hours the patrol force must expend $\lambda t/\mu$ patrol car hours servicing calls. If N cars are fielded we have Nh hours of work time so that A, the hours available for preventive patrol, is given by

$$A = Nh - \lambda t/\mu. \qquad (7)$$

We assume that A is greater than zero. If it were not, all calls could not be handled, very large queues would develop and the situation would be generally out of hand.

Clearly, according to (7), every car added increases A by h hours. (This makes a tacit assumption that the behavior of the patrol force is constant.) If this is not so, and there is more time out when more cars are fielded, for example, then h is effectively decreasing and (7) does not hold. A little reflection on (7) will indicate that when many cars are answering calls, A is small and small percentage increases in N can yield enormous percentage increases in A. We quantify this relationship as follows: Suppose we wish to increase A by a factor δ (that is, by 100δ percent). What is β, the associated proportional increase required by N? From (7) we see that the relation is

$$(1 + \delta)A = (1 + \beta)\, Nh - \lambda t/\mu \tag{8}$$

or

$$\beta = \delta \left(1 - \frac{\lambda t}{\mu h N}\right). \tag{9}$$

The term $\lambda t/\mu hN$ is the average proportion of work time that a car uses in responding to calls for service. It is clear that when this proportion is high we get very significant percentage increases in A for small percentage increases in N.

We take some numbers from the sample problem to illustrate this effect. The average total car hours ($\lambda t/\mu$) needed to service calls over the midnight to 8 a.m. tour is about 29. In the actual schedule, about six cars are fielded. Ignoring any other nonservice-nonpatrol activity save meals, this gives 42 available work hours, or on the average about 13 hours available for patrol. To increase the patrol hours by 100 percent to 26 hours would require 13/7 = 1.86 more cars, an increase of only about 31 percent over the six cars assigned. The numbers are only meant to illustrate these effects and are not intended to indicate the desirability or undesirability of any actions.

Probability That All Cars are Busy

We now determine how the number of patrol cars required depends on α, the probability that all cars are busy. This is much more difficult to study than availability for preventive patrol, which was just discussed, because α is linked to the number of cars by complicated queueing equations that we can solve only by numerical integration. Moreover, the resulting relationship is fundamentally nonlinear--in contrast to (7).

An approximate but essentially accurate impression of this relationship can be obtained by examining some numbers produced by the M/M/n/n queueing model that is used to generate the requirements for the scheduling linear programs. We used this model to generate hour-by-hour car requirements for various values of α. In Fig. 6 we display these results for $\alpha = 0.50$ and 0.01. The nonlinearity of the relationship is apparent. The most important aspect of this nonlinearity is that each additional car results in smaller and smaller decreases in α. Figure 7 illustrates this more concretely. Here we have plotted α versus the number of cars on duty for a specific value of $\rho \equiv \lambda/\mu$, namely $\rho = 5.0$, which is about the highest hourly demand rate in the 71st Precinct. A minimum of six cars must be assigned or extraordinary queues will develop. With six cars $\alpha = 0.59$, and by adding a seventh car α is reduced by 0.27 to 0.32. Adding an eighth car gives a smaller reduction of 0.15 to 0.17, and so on until adding a twelfth car yields a reduction of only 0.01 below the value of α for eleven cars. The phenomenon of decreasing returns is quite general --it holds for all parameter values.

Results obtained from the scheduling model also illustrate this phenomenon. Figure 8 plots the number of cars vs. α for computer-generated schedules. Two types of schedules are shown:

(1) Schedules using the standard three tours and the standard meal times during the middle four hours of each tour.

(2) Schedules using four tours, the standard ones and an additional tour starting at 2000 hours, and permitting a meal break at any hour of the tour.

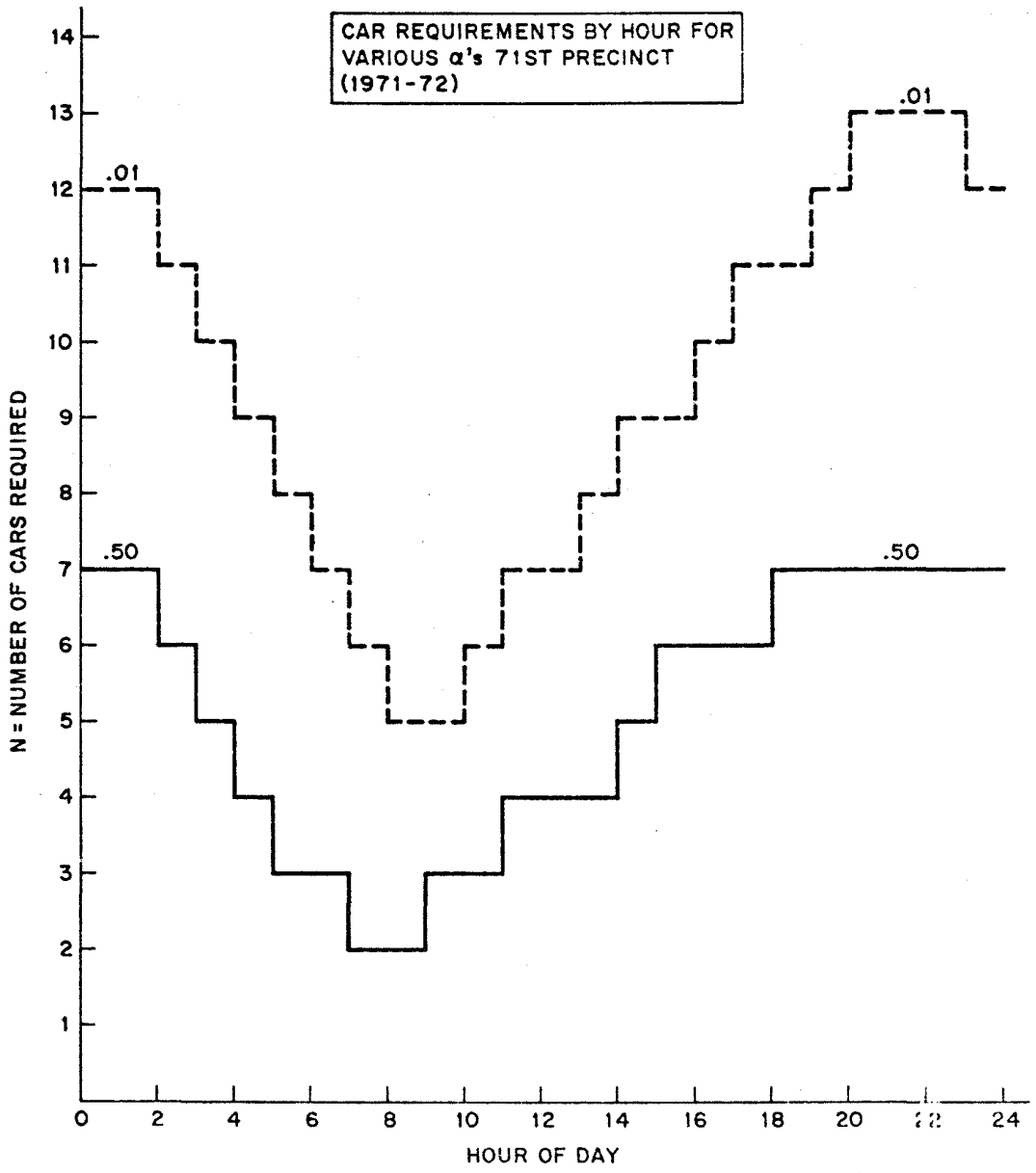

Fig. 6. Car requirements by hour for $\alpha = .01$ and $\alpha = .50$

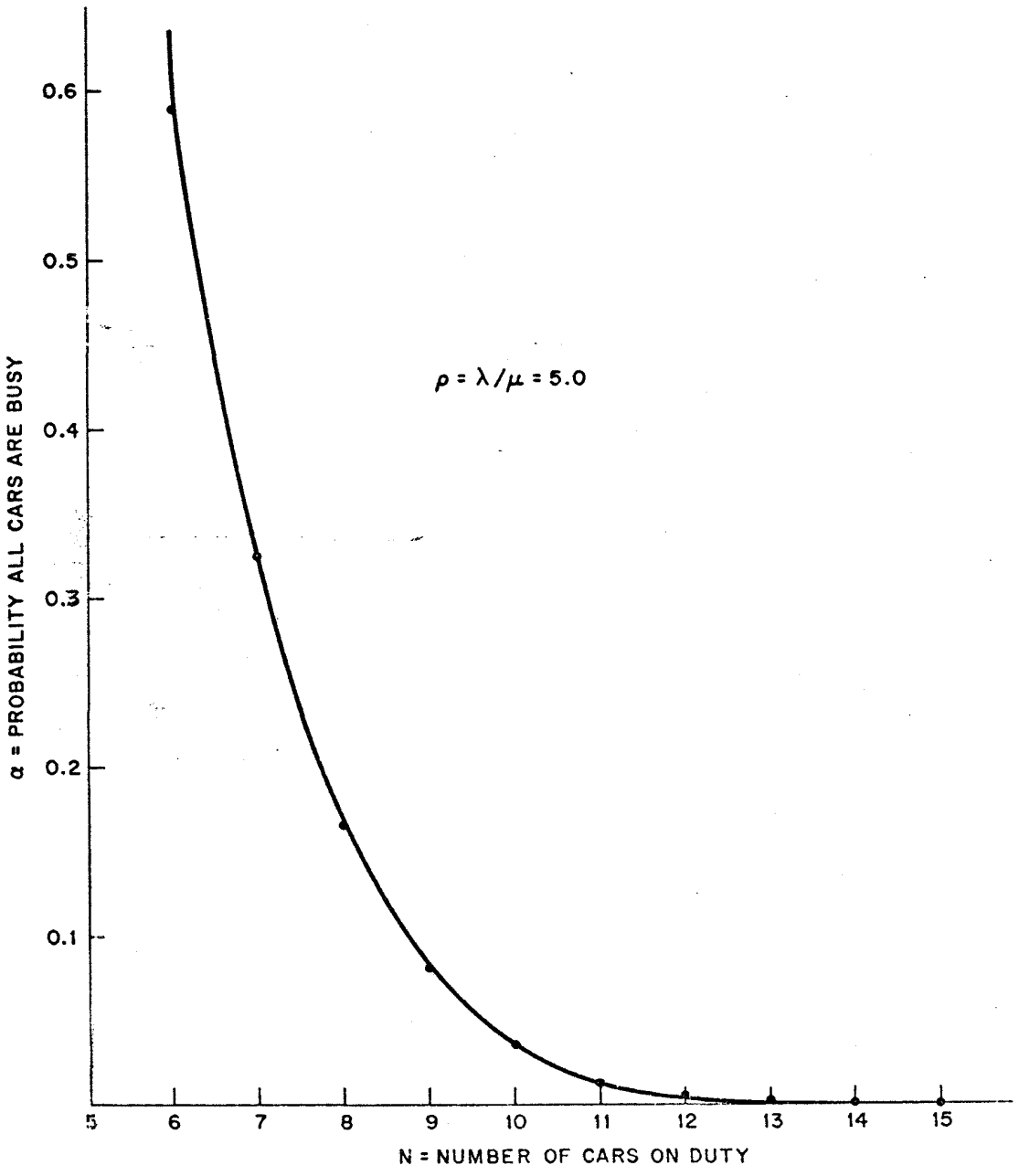

Fig. 7. Probability that all are cars busy vs. the number of cars on duty

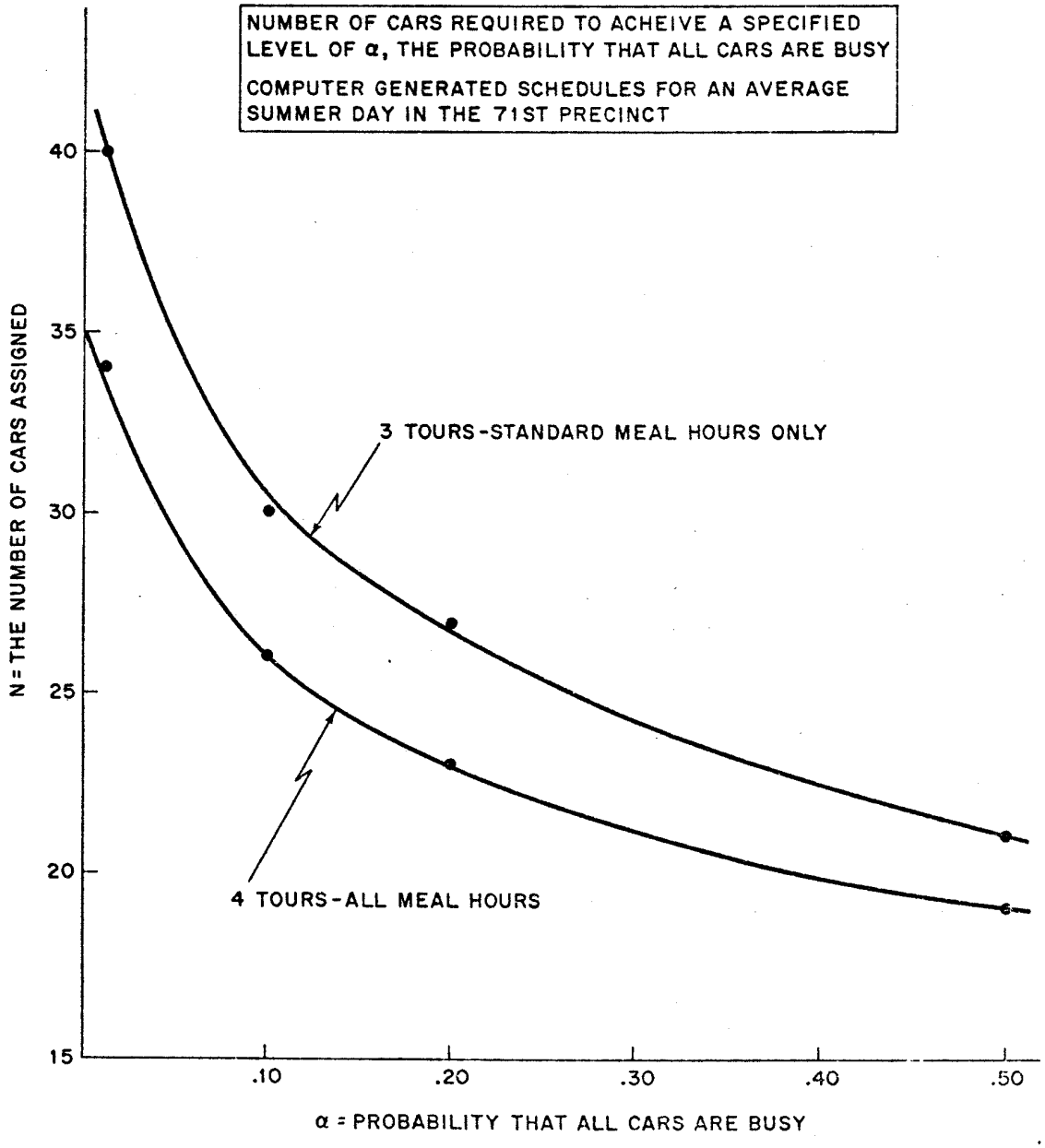

Fig. 8. The number of cars assigned vs. the probability that all cars are busy

The curves for both tour configurations show the same decreasing convex relationship

SUMMARY OF THE RELATIONSHIP BETWEEN PERFORMANCE AND THE NUMBER OF CARS FIELDED

An overview of two relationships--A, car hours available for patrol and α, the probability that all cars are busy as functions of N, the number of cars assigned--has already been shown in Fig. 5. There we plot A and $1 - \alpha$, the probability that at least one car is available, vs. N for the same two types of schedules that were just discussed (and whose results are given in Fig. 8). We see, of course, that both A and $1 - \alpha$ increase with N, and that $1 - \alpha$ increases at a decreasing rate while the number of car hours available for patrol increase linearly.

Two important observations can be made:

- The number of car hours available for patrol are the same for either--in fact for any--schedule with the same number of cars assigned. The schedule only affects how these hours are distributed during the day.
- With respect to $1 - \alpha$, the four-tour schedule with no restrictions on meal hours is superior to the three-tour schedule with standard meal hours for any number of cars assigned. Further, the difference is largest in the range (about 24 cars) in which the department was operating during the period when the data were gathered.

SCHEDULING OVER THE WEEK

All of the analyses and results presented so far have been for scheduling over a single--typical--day. A great deal about scheduling can be learned by such analysis, but one must also explicitly face the fact that demand for police patrol service varies over the week.

There are small variations from day to day. These variations are not large enough to require daily differences in schedules. But variations large enough to require different schedules do occur between weekdays and weekends. These differences are primarily during the midnight to 8 a.m. period of Saturday and Sunday. (See Table 3 which gives the hourly call rates by day of week for the 71st Precinct for the year from September 1971 to October 1972.) The weekend-weekday variations are shown in Fig. 9.

Table 3

CALL RATE BY HOUR AND DAY OF WEEK

(71st Precinct, Brooklyn, September 1971-October 1972)

Hour	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
1	8.86	5.56	5.38	5.94	6.08	5.54	7.63
2	8.56	4.65	4.71	4.48	4.63	5.29	7.25
3	7.46	3.58	3.50	3.42	3.60	3.23	6.48
4	6.96	2.21	2.62	2.48	2.35	3.12	5.15
5	5.37	2.06	1.85	1.96	1.58	2.20	4.50
6	4.21	1.94	1.52	1.42	1.77	1.98	3.40
7	3.10	2.42	2.25	1.71	2.17	2.38	3.27
8	2.77	3.20	2.98	2.35	2.73	3.48	2.96
9	3.81	5.04	4.38	4.35	4.69	5.31	4.21
10	4.42	5.71	5.29	4.46	5.23	4.69	5.27
11	4.52	5.27	5.38	4.52	5.04	5.17	5.63
12	5.15	5.42	5.52	4.88	4.88	5.92	5.42
13	6.00	6.23	5.40	5.44	5.46	5.62	5.06
14	5.67	5.98	6.42	5.69	5.55	6.12	6.48
15	5.54	6.37	5.54	5.75	6.04	5.48	5.88
16	6.23	7.58	5.62	6.25	6.04	6.62	6.23
17	6.50	7.75	6.70	6.54	7.65	7.35	7.19
18	7.04	8.10	7.65	7.60	8.56	7.75	8.37
19	6.63	8.85	9.44	8.92	8.83	7.90	8.08
20	7.67	8.29	8.04	8.56	8.10	9.19	8.80
21	7.70	8.17	8.10	8.96	7.73	7.73	9.10
22	7.62	8.52	7.67	7.85	8.25	7.71	8.00
23	7.83	7.87	7.38	7.38	7.73	7.17	7.77
24	6.27	6.42	6.37	6.77	6.65	7.02	7.92

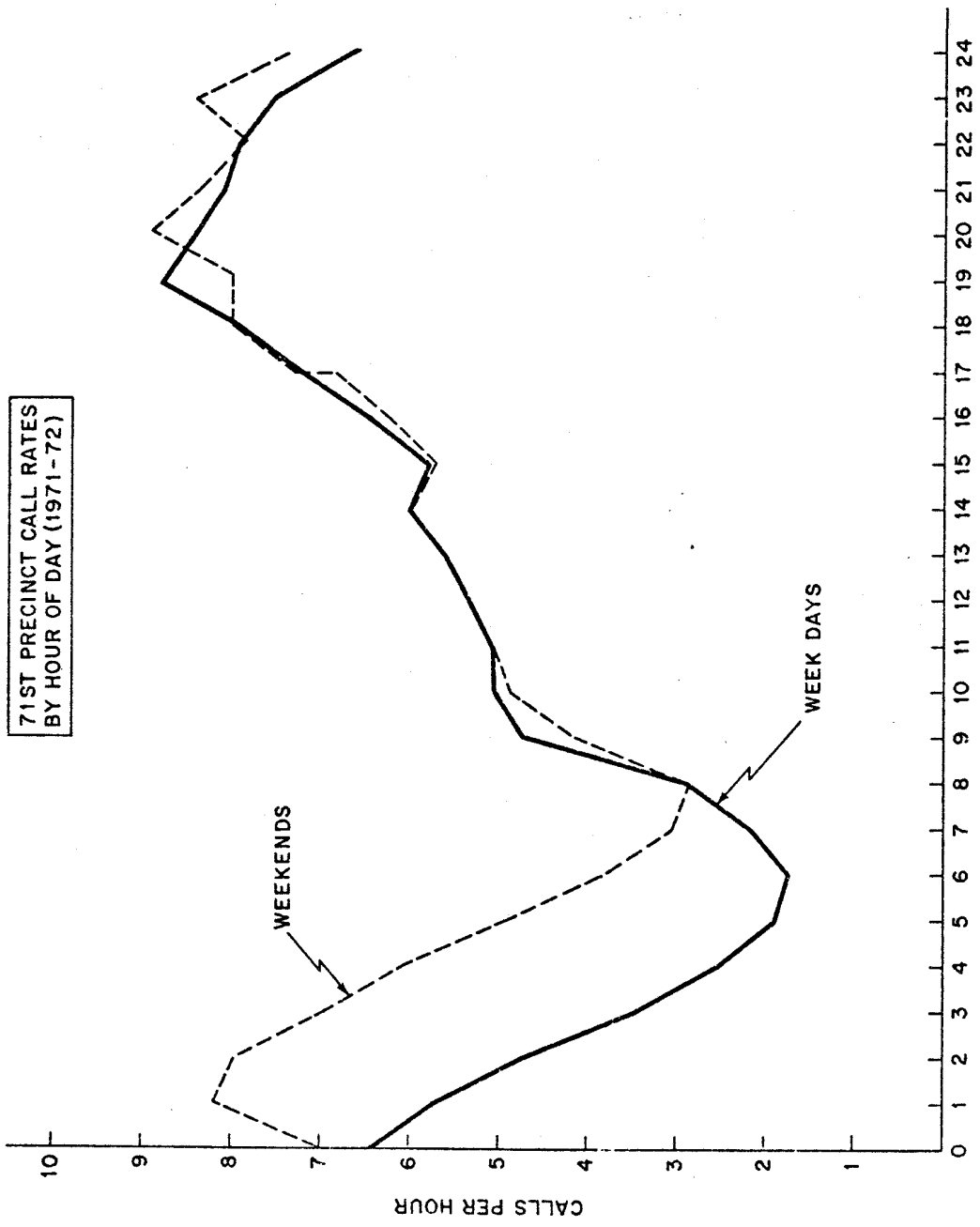

Fig. 9. Calls by hour of the day in the 71st Precinct, 1971-1972

We determined the differences in schedules between weekends and weekdays by using a linear programming model for the entire week. The model includes all 168 hours of the week. Our analyses used hour-by-hour call rates averaged over all weekdays and averaged over both Saturday and Sunday for the weekend. The data spanned an entire year.

Several types of schedules were generated, including schedules using the three standard tours, four-tour schedules using the three standard tours, and a four-tour schedule starting at 2000 hours. Schedules were generated with both the standard meal breaks and with meal breaks permitted during any hour of the tour. In addition to schedules generated under the assumption that all weekdays are the same and both weekend days are the same, we generated schedules for which the call rate pattern was different for each day of the week.

The results of the analysis can be summarized simply:

- The advantages of four-tour and five-tour schedules that were generated for a "typical day" also hold for schedules over an entire week.
- The advantages of flexible meal hours in schedules generated for a "typical day" also hold for schedules over an entire week.
- The weekdays and weekends are only "loosely coupled." That is, the additional cars required on duty during the early morning hours on the weekends are obtained by simply scheduling more cars during the tours starting at 0000 hours on those days. The weekday schedules are the same as those generated if a typical weekday is considered as a separate scheduling problem. The weekend schedules are the same as those generated if a typical weekend day is regarded as a separate scheduling problem.

Table 4 displays a week-long schedule for the 71st Precinct using four tours with $\alpha = .1$ and with meal breaks permitted during all hours. Note that each weekday a total of 21 cars are assigned, with 4, 7, 8, and 2 cars, respectively, being allocated to each tour. The weekend days have 23 cars assigned, the additional two cars going to the first tour (midnight to 8 a.m.), making the pattern 6, 7, 8, and 2.

Table 4

NUMBER OF PATROL CARS ASSIGNED DURING THE WEEK--
A COMPUTER GENERATED SCHEDULE[a]

Tour Start Time	Sun.	Mon.	Tues.	Wed.	Thur.	Fri.	Sat.
0000	6	4	4	4	4	4	6
0800	7	7	7	7	7	7	7
1600	8	8	8	8	8	8	8
2000	2	2	2	2	2	2	2

[a] $\alpha = 0.10$, meal breaks permitted during all hours.

DEMAND PATTERNS IN DIFFERENT PRECINCTS

Most of our analyses were based on data from the 71st Precinct in Brooklyn. If this precinct had a pattern of calls for service that was atypical--by hour of day, or day of week, etc.--it would be erroneous to apply the scheduling principles just discussed to other precincts. We therefore carried out an extensive analysis of demand patterns in six precincts that were suggested by the NYPD as having a broad range of characteristics. The precincts studied are identified in Table 5. For these precincts we studied the time patterns of service times and of calls for service--by hour of day, by day of week, and by month. We found that for all precincts, the assumption that the number of calls per hour follows a time-varying Poisson process is quite accurate. Moreover, we found that the patterns by hour of day are very similar. Figure 10 illustrates these results.

We conclude that although there certainly will be individual exceptions, the scheduling principles outlined here should apply to most police precincts in New York City.

Table 5

PRECINCTS STUDIED IN DEMAND ANALYSIS

Precinct	Location
13	Lower Manhattan
44	South Bronx
66	Central Brooklyn
71	Heights, Brooklyn
83	Central Brooklyn
103	Eastern Queens

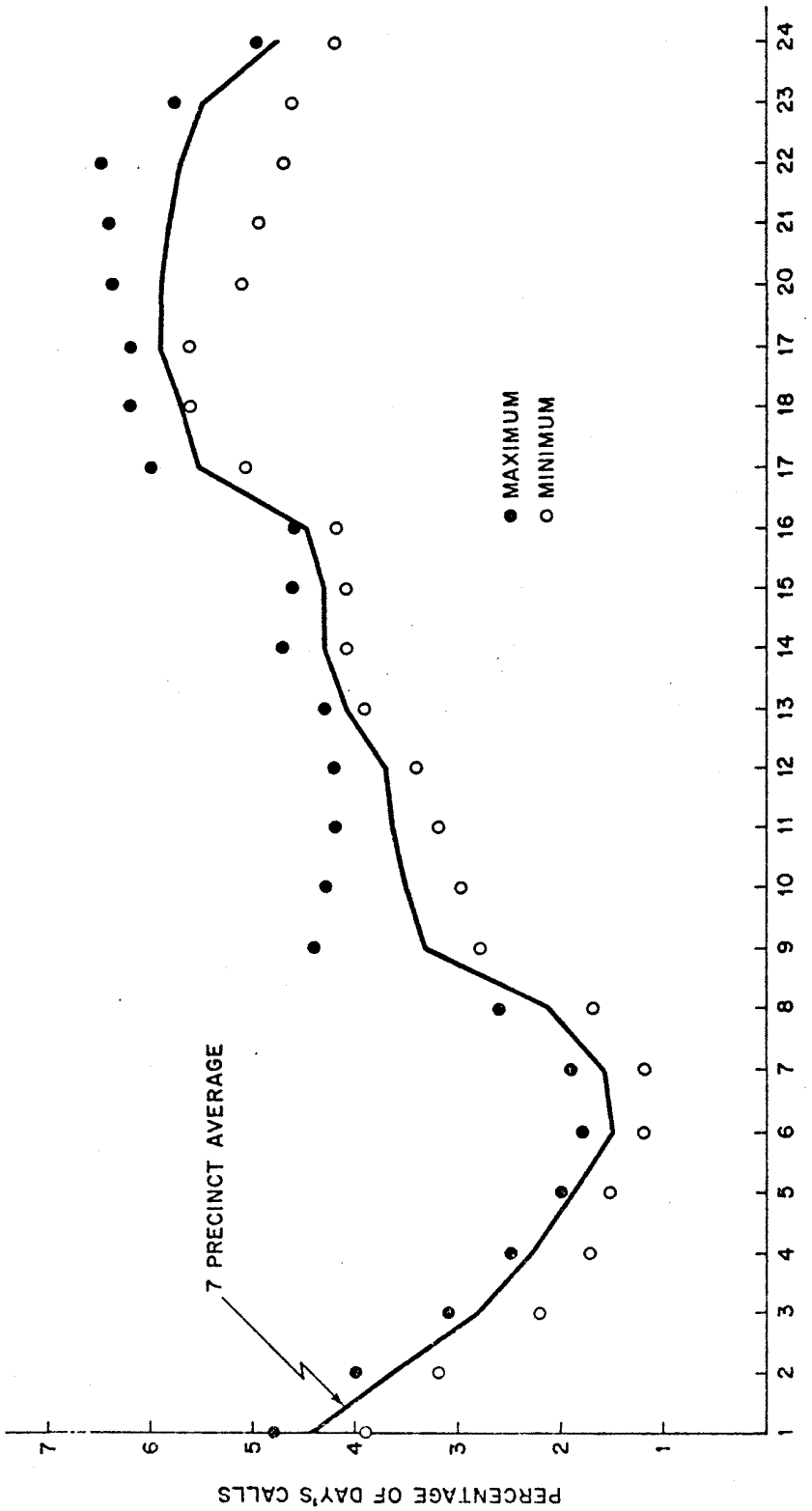

Fig. 10. Hourly call rate pattern in seven precincts

APPENDIX A

An IBM program product [14], the Mathematical Programming System Extended (MPSX) with Mixed Integer Programming (MIP), was used to obtain integer solutions to the scheduling problem. MIP uses the branch and bound method of solving mixed integer programming problems, and allows the user to choose a standard solution strategy or to implement a strategy appropriate to the structure of the problem.

We modified the standard solution strategy because of several special features of the scheduling problem. In our problems, all feasible integer solutions have integer-valued objective functions, and the optimal integer solution is, in general, very close in value to the optimal solution to the problem when integer constraints are relaxed. Also, there may be many alternative optimal solutions, but for our purposes, we often do not need to enumerate these alternatives. Finally, the standard MIP strategy makes heavy use of the "pseudocosts" of altering variable values and of the variable weights in the objective function. Since all of the integer variables in our objective function have the coefficient unity, some modification of the branching rules seemed desirable.

When the search for an optimal integer solution commences, there is already available an optimal solution to the problem obtained by relaxing the integer constraints. We take advantage of the fact that the integer solution is often close to this "continuous solution" by placing all nodes whose objective function value is at least four cars more than the continuous solution into an inactive state. If there is no integer solution this close to the continuous solution, a new set of nodes with values of no more than eight cars higher than the continuous solution is amde active. This process continues until an integer solution is found. In practice, an integer solution has always been found among the first set of nodes.

When a feasible integer solution is found, all nodes with a value lower by less than one are placed in an inactive state and all nodes with values worse than the solution are dropped. Because we know that any integer solution must have an integer functional value, we know that the nodes

just made inactive cannot lead to an integer solution better than that just obtained. The nodes are not dropped, however, since the last solution might be optimal and we may want alternative optimy for some problems.

As the search progresses, the best functional value among all the nodes, which is usually noninteger, will become larger. If we obtain an integer solution that has a functional value equal to the best functional value rounded up to an integer, we know that the solution is optimal. If only one optimal solution is desired, the search can stop at this point. If alternative optimy are wanted, the search continues.

During the search process the variable expected to give the greatest expected functional deterioration is chosen as the branching variable, and all variables with current values $[x]+.2<x<[x+1]-.2$ are given priority for branching consideration. The node with the best functional value is chosen in order to obtain a "bushy" tree [1]. Since we know that the optimal integer solution will usually be close to the optimal continuous solution and since we also know that the optimal solution can be no smaller than the best functional value rounded to the next highest integer, this procedure ensures a quicker "proof of optimality" even though it may take longer to find the first integer solution.

While the standard solution strategy would have been sufficient to obtain the desired solutions, the revised strategy allows a quicker proof of optimality and a more predictable behavior of the search process for our particular problem.

APPENDIX B

An IBM program product, the Continuous System Modeling Program (CSMP) [3], was used in the numerical solution of the time-dependent queueing equations. CSMP permits description of the queueing model with FORTRAN-like statements, and the user can select among various numerical integration techniques.

We selected the fourth-order Runge-Kutta method with a variable step size. Two difficulties peculiar to the family of equations we were solving led to this choice. First, Runge-Kutta was chosen because there is a discontinuity in the equations at each tour change or scheduled meal. There are many numerical integration methods that evaluate the equations to be integrated several steps ahead and use the results to estimate higher-order derivatives, which are in turn used to accelerate the integration.* Unfortunately, the estimated derivatives are incorrect when a discontinuity exists in the equations. The Runge-Kutta techniques, in contrast, concentrate on a single step and do not look ahead. Therefore, if a single step does not pass over a shift change, there will be no discontinuity in the calculations.

Second, a variable step size was chosen because after a shift change there may be a quick transient response in the solution. The intervals over which strong transient responses occur constitute only a small fraction of the entire interval of integration. A fixed-step procedure would have required an uneconomically small step size over the entire time period of the calculations to insure the numerical stability of the solution immediately after the shift changes. Therefore a variable step procedure was chosen with the constraint that a step could not straddle a shift change.

During the numerical evaluation of the equations, the state probabilities were constrained to be greater than or equal to zero. A check was made of numerical accuracy by summing the probabilities. At no point in the solution did the sum deviate from unity by more than 0.0014.

*See [12] for a discussion of alternative numerical integration techniques.

As initial values for the integration, we used the steady state solution for the M/M/n system with the midnight parameter values. The integrations were run for a two-day interval and the solutions for the first day were compared to the solutions for the second day. After a sufficiently long time the effects of the initialdistribution should disappear and one might expect that two solutions of the equations separated by 24 hours would be close. In all cases, the probabilities converged to the periodic solution well before the onset of the second day. Of course, we did not learn this until the second day had been solved. The values of $p_j(t)$ for the second day were then used as the periodic solutions to the equations.

REFERENCES

1. Claude Berge, The Theory of Graphs, John Wiley and Sons, New York, 1962.

2. J. Chaiken, Patrol Allocation Methodology for Police Departments, R-1852-HUD, The Rand Corporation, September 1975.

3. "Continuous System Modeling Program," IBM Program Product, Program Number 360A-CX-16X, 1972.

4. R. Garfinkel and G. Nemhauser, Integer Programming, John Wiley and Sons, 1973.

5. S. Gass, J. Dawson et al., "An Evaluation of Policy-Related Research: Reviews and Critical Discussions of Policy-Related Research in the Field of Police Protection," Mathematica, Inc., Bethesda, Maryland, 1974.

6. A. Geoffrion and R. Marsten, "Integer Programming Algorithms: A Framework and State-of-the-Art Survey," Management Science, 18, 1972, pp. 137-163.

7. N. B. Heller and W. W. Stenzel, "Design of Police Work Schedules," Urban Analysis, 2, 1974, pp. 21-49.

8. T. C. Hu, Integer Programming and Network Flaws, Addison Wesley, Reading, Mass., 1971.

9. E. Ignall, P. Kolesar, and W. Walker, "The Use of Simulation in the Development and Empirical Validation of Analytic Models for Emergency Services," Proceedings of the 1974 Winter Simulation Conference, Washington, D.C., 1974, pp. 529-540.

10. P. Kolesar and W. Walker, A Simulation Model of Police Patrol Operations: Program Description, R-1625/2-HUD/NYC, The Rand Corporation, February 1975.

11. B. O. Koopman, "Air Terminal Queues Under Time Dependent Conditions," Operations Research, 20, 1972, pp. 1089-1114.

12. L. Lapidus and J. H. Seinfeld, Numerical Solution of Ordinary Differential Equations, Academic Press, New York, 1971.

13. R. C. Larson, Urban Police Patrol Analysis, The MIT Press, Cambridge, Mass., 1972.

14. "Mathematical Programming System Extended (MPSX-MIP)," IBM Program Product, Program Number 5734-XM4, 1973.