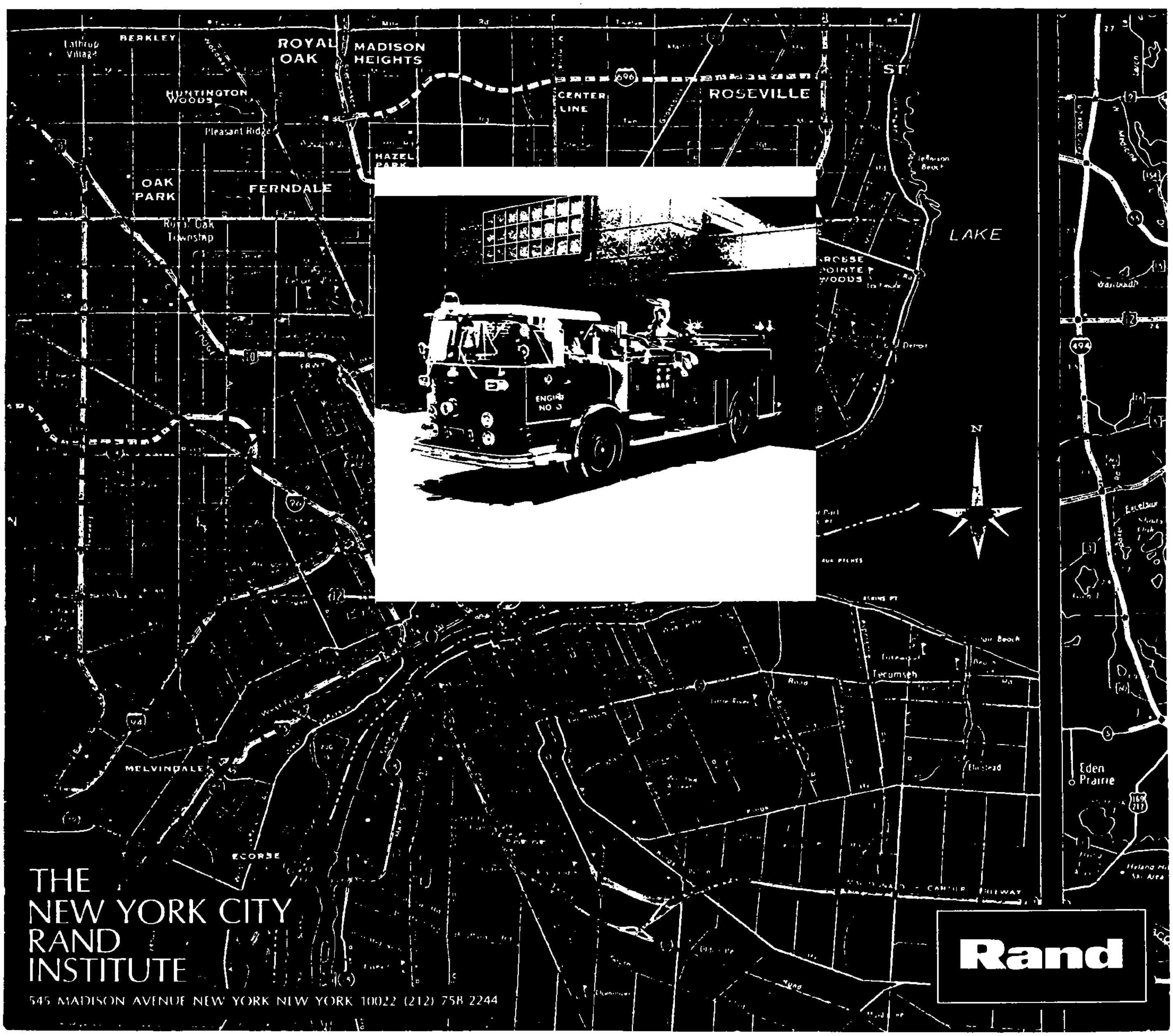
FIREHOUSE SITE EVALUATION MODEL: DESCRIPTION AND USER'S MANUAL

PREPARED FOR THE DEPARTMENT OF HOUSING AND URBAN DEVELOPMENT

PETER DORMONT JACK HAUSNER
WARREN WALKER

R-1618/2-HUD
JUNE 1975



THE NEW YORK CITY RAND INSTITUTE
545 MADISON AVENUE NEW YORK NEW YORK 10022 (212) 758 2244

Rand

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. Neither the United States nor HUD, nor any person acting on behalf of HUD, 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.

The New York City-Rand Institute is a nonprofit research institution 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 administration 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 techniques 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

FIREHOUSE SITE EVALUATION MODEL: DESCRIPTION AND USER'S MANUAL

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

PETER DORMONT
JACK HAUSNER
WARREN WALKER

R-1618/2-HUD
JUNE 1975

THE
NEW YORK CITY
RAND
INSTITUTE
545 MADISON AVENUE NEW YORK NEW YORK 10022 (212) 758-2244

Rand

PREFACE

This report was prepared to provide fire department planners, systems analysts, and data processing personnel with detailed documentation of a computer-based model, called the Firehouse Site Evaluation Model, that can be used to evaluate alternative configurations of fire-fighting companies. The model was developed and tested in several cities under a contract with the Office of Policy Development and Research of the United States Department of Housing and Urban Development (HUD). The objective of the contract is to develop, test, and document 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.

The Firehouse Site Evaluation Model is one of several models that have been developed by The New York City—Rand Institute to help in deciding how many fire companies should be on duty and where they should be located. The Introduction to this report provides information that can be used to determine which of the models is most appropriate for a particular application. Additional information is provided in a companion report that describes the Firehouse Site Evaluation Model and its uses for fire department administrators and other city officials:

> R-1618/1-HUD, *Firehouse Site Evaluation Model: Executive Summary*, Warren E. Walker, The New York City—Rand Institute, April 1975.

These two reports are part of a series being produced under the HUD contract that document several different deployment models for police, fire, and ambulance services. A second series of reports describes the application of the models in several cities. A third set of reports will present the general methodology developed for emergency service deployment analysis. It is hoped that these reports will provide local government officials with methods for systematically assessing alternative policies for providing emergency services, and will enable them to conduct deployment analyses with little or no outside technical assistance.

The HUD contract with The New York City—Rand Institute is one of the efforts supported under HUD's Community Development and Management Research

Program. The Program is designed to develop, field test, and provide to state and local officials new approaches and methods for responsive community management. The methods are designed to help decisionmakers identify alternative policies and actions and assess the feasibility, costs, and consequences of these alternatives. Methods are tested in representative communities under actual operating conditions. The methods and test results are made available to users in other communities.

SUMMARY

This report describes a computer-based model, called the Firehouse Site Evaluation Model (or "siting model"), that can be used to evaluate alternative configurations of fire-fighting companies. The siting model provides a way to estimate the fire protection levels, measured in terms of travel times, travel distances, and company workloads, that would result from implementation of any given arrangement of fire companies. By comparing the fire protection levels resulting from one arrangement with those resulting from others, rational decisions can be made about the deployment of a city's fire companies.

The report is organized so that persons with different interests in the siting model can read the appropriate sections without having to read the entire report. The earlier sections are designed to be read by fire department planners and systems analysts. They provide information about how and when to use the model, the model's assumptions, and the input to and output from the model. An illustration of how the model was used to analyze fire company deployment in Trenton, New Jersey is also included. The later sections of the report are designed for the data processing personnel who will be installing and maintaining the siting model. They include a program listing, a description of the input data, sample printouts, and detailed instructions for installing the model.

ACKNOWLEDGMENTS

The Firehouse Site Evaluation Model was developed and refined at The New York City—Rand Institute over the course of several years. During this period it was used in performing fire company deployment analyses in: Yonkers, New York; Trenton and Jersey City, New Jersey; and Wilmington, Delaware. We gratefully acknowledge the helpful comments and suggestions that we received from the civilian and uniformed personnel who worked with us in these cities. Mei Ling, a member of the Institute's staff, programmed an earlier version of the model.

We are grateful to Herbert Shukiar of The Rand Corporation and Robert Baumgardner of the U.S. Department of Housing and Urban Development, who read an earlier draft of this report and provided the authors with suggestions that have served to make the report more understandable and readable.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Fire department managers, confronted with new and greater demands for fire service and fire protection, require new tools for monitoring and re-evaluating the fire protection needs of their cities. The dynamics of demographic change and the resulting change in fire experience often pose severe challenges to the planning function of the fire service. In addition, the steadily rising operating costs of the fire service require the effective and efficient deployment of fire-fighting resources. This report describes a computer-based model that can be used by fire administrators and city planners to evaluate the effects of contemplated changes in the deployment of fire-fighting resources in a city.

The determination of a configuration of fire companies for a city involves obtaining the answers to two strategic deployment questions*:

(1) How many fire companies does the city need in order to provide a certain level of fire protection?

(2) Where should the fire companies be located?

Obtaining answers to these questions involves subjective judgments and trade-offs among competing objectives. However, several mathematical models have been developed to provide the decisionmaker with quantitative information, which he can combine with judgments on the non-quantifiable aspects of the problem in order to reach a final decision. These models fit into a general methodology for analyzing fire company deployment problems, which is documented in [1].**

One of the models, the Parametric Allocation Model [2], provides the user with a general picture of the number of fire companies needed in different parts of the city and the fire protection levels that could be obtained with different numbers of companies. It is very quick and inexpensive to use and requires very little data to be collected, but cannot be used to evaluate specific locations for the companies. Its primary purpose is for assistance in the initial steps of a study.

The Parametric Allocation Model can be used to assess whether or not a serious problem exists in the current distribution of fire companies throughout a city by providing information that can be used to compare fire protection

---

*See "Deployment" in Appendix A (Glossary) for a discussion of fire company deployment issues.

**Figures in square brackets identify references listed at the end of this document.

levels among regions of the city. If imbalances are found, the model can also be used to determine how to reallocate the existing units among the regions to provide more balanced fire protection. If fire companies are to be added to, or eliminated from the fire department, the model can also be used to determine the regions that should gain or lose the companies.

Once a fire department has this information, it is not difficult to develop several alternative configurations of station sites that might lead to improved performance. These can be evaluated in detail using the Firehouse Site Evaluation Model (also called the siting model) described in this report, or the Fire Operations Simulation Model described elsewhere [3,4]. These models require substantially more elaborate data than the Parametric Allocation Model.

The simulation model is a complex computer program that can be used only by persons who understand the special programming language used; it also requires a moderately large computer, is expensive to operate, and requires a very considerable amount of data as input. It should be used in preference to the siting model only if the greater capabilities and better accuracy of the simulation model are required for the analysis of the deployment policies being considered. In particular, the simulation model is to be preferred to the siting model if the alarm rate in the city is high relative to the number of companies so that it is reasonably likely that an alarm will occur in a company's response area while it is busy at another alarm.

The siting model assumes that companies will almost always be available in their firehouses to respond to alarms. This is a reasonable assumption for fire departments in most cities, since fire companies typically spend less than five percent of their time responding to and working at alarms. The model provides a way to estimate what fire protection levels would result from implementation of a specific arrangement of firehouses. By comparing the fire protection levels resulting from one arrangement to those resulting from others, a fire department can make rational decisions about the location of its fire companies.

The model describes the effects of a given fire company configuration in terms of a wide variety of performance measures. These include estimates of:

(1) The resulting average travel times in each region and resulting travel times to specific hazardous locations.

(2) The frequency distribution of travel times to potential incident locations within a region, including the maximum travel time from a station to any location in that region.

(3) The resulting workload of each company, measured by the historical number of alarms that have occurred in the company's response area.

The siting model is applied iteratively. First, the locations of alarm boxes and the locations and response areas of the existing fire companies are described in a form that the computer program can read. The model then calculates the travel-time and workload characteristics of the existing arrangement. From this it might be seen that one fire company has too much work to do as compared to the others, or that travel times are unacceptably high in one part of the city. This information will suggest places to add, delete, or move firehouses. The model then can be used to provide information on a new arrangement, which might indicate still further changes. This process continues until a satisfactory arrangement is achieved.

The siting model does not by itself generate alternative firehouse configurations. It simply helps the fire department to evaluate alternative configurations that their planning personnel create. What the model provides is detailed quantitative information about each configuration, thereby permitting a structured, rational analysis to select the best ones. In selecting actual sites, this information is combined with subjective judgments about other considerations, such as fire hazards.

The Firehouse Site Evaluation Model is an interactive computer program that can be accessed via a portable terminal or teletype. This allows the user to formulate deployment questions and obtain the results while sitting in his own office.* It has been used in the analysis of fire company deployment policies in Trenton, New Jersey [5]; Yonkers, New York [6]; Jersey City, New Jersey [7]; and Wilmington, Delaware [8]. The Trenton study is used as an example in Section 6 to illustrate the use of the model.

This report serves as an introduction to the model as well as a user's manual for the computer program, and is organized so that persons with different interests in the siting model can read the appropriate sections without having

*The program is written so that it can also be used off-line in a "batch" mode, but this report is written as if the user will be using the model interactively at a computer terminal. Instructions for setting up and using the model in batch mode are given in Section 8.6.

to read the entire report. The analyst or decisionmaker should read Section 2, which explains how the model can be used to evaluate a city's fire protection; Section 3, which explains the assumptions on which the model is based; Section 4, which explains the model's output; Section 5, which explains the commands that are used to communicate with the model; and Section 6, which illustrates how the model was used to analyze fire company deployment in Trenton, New Jersey. Once the user is familiar with the model he will generally need to refer only to Sections 4 and 5. The data processing personnel who will have the responsibility of installing the model and creating the city's data base need only read Section 7, which describes the model's data requirements, and Section 8, which provides detailed instructions for installing the model on the user's computer system. A glossary of terms used in this report that may be unfamiliar to the reader is provided in Appendix A.

It must be emphasized that the siting model is purely descriptive. It does not recommend a configuration; it only estimates what would happen if a given configuration of fire companies existed. Its use requires extensive user interaction. The results from the model should be regarded as an aid to analysis rather than as a substitute for it. In the final analysis, it is a matter of managerial and analytic judgment to choose from among several alternative configurations the one that best meets the fire protection needs of the city while remaining within budgetary limits.

2. EVALUATING A CITY'S FIRE PROTECTION

2.1. PERFORMANCE MEASURES

Ideally, we would like to measure the effectiveness of a given deployment of fire-fighting resources in terms of its effect on the safety of life and property, but we cannot determine the effect that a deployment policy will have on these measures because the relationship between the loss of life and property and the deployment of fire-fighting units is not known. However, the siting model can be used to calculate several surrogate measures of fire protection whose values we can estimate and which, in many cases, are directly related to loss of life and property. These measures fall into three categories:

(1) travel times to specific locations;
(2) average travel times; and
(3) company workload.

Changing the deployment of fire companies in a city will result in changes in these measures. For example, adding fire companies to a region will reduce the average travel times in the region and will reduce the workloads of the companies in the region. An analysis of these measures, together with judgments concerning hazards to life and property, projected alarm patterns, and existing political, economic, and social constraints, can be used to evaluate the deployment changes being considered. We discuss each of these sets of measures below. The siting model calculates these measures separately for each type of fire-fighting company (engines, ladders, etc.).

2.1.1. Travel Times to Specific Locations

In most cities, fire companies spend less than five percent of their time fighting fires. At most other times they are performing a different function: providing insurance against fire loss. That is, they are available in case a fire occurs and are situated so that they can respond quickly to alarms. The protection provided by a fire department to any given location can be measured in terms of the distance from that location to the nearest firehouse or, more importantly, the expected amount of time it would take a fire company to travel to that location. Locations close to a firehouse are well protected; those far from a firehouse are more poorly protected. The siting model estimates the travel time from the closest firehouse to every point identified as a potential fire site in a given region (implicitly assuming that the fire company is available in the house to respond) and produces a frequency distribution of the travel times. In addition, it lists for the region the maximum travel time from the nearest firehouse to any potential incident location in the region. The travel times can also be calculated (at the user's option) from the second closest firehouse, third closest, etc.

The potential fire sites can correspond to street alarm boxes, grid coordinates, street intersections, etc. In the rest of this report (and in the output from the model) they will be referred to, for convenience, as alarm boxes.*

2.1.2. Average Travel Times

Averaging the estimated travel times to all alarm boxes in a region produces another useful measure of the fire protection being provided in the region. The model calculates both unweighted and weighted average travel times. The unweighted average gives equal weight to all boxes in the region, regardless of how alarms are distributed throughout the region. The weighted average takes into account the fire experience at each box in calculating the average travel time. It weights the travel distance to each box by the total number of incidents (or, alternatively, structural fires) that occurred at that box in a given year. The resulting average is a good estimate of how soon the fire department can be expected to arrive at an average incident (or structural fire) that occurs in the region. The regional average travel times can be calculated (at the user's option) for second-due companies, third-due, etc.**

2.1.3. Company Workload

This measure, although not directly related to the loss of life and property, can be important in cities where the alarm rate is high. An overworked company may not be as consistently effective as a company with a lower workload. In addition, it may be important to maintain a reasonably balanced

*For a more complete definition, see "Alarm Boxes" in Appendix A.

**See the definition of "First-Due Company" in Appendix A.

workload among all fire companies. The workload of an individual company is measured by the total number of incidents and the number of structural fires that can be expected to occur annually in the area in which it has the responsibility of first response, i.e., its "first-due" area.* Company workloads can also be calculated (at the user's option) for "second-due" areas, "third-due" areas, etc.

2.2. DEFINING THE REGIONS

The measures described above can be used to evaluate any change in the configuration of fire companies in a city. However, one must be careful in defining the regions for which these measures are to be calculated. The change in the citywide average travel time that would result from moving (or even eliminating) a single company can be extremely small. For example, in a city of about twenty square miles with fifteen fire companies, the removal of one company might increase citywide average travel time by about five seconds. But use of this citywide average would give an erroneous impression of the true magnitude of the change, since a citywide increase of five seconds in average travel time may well result from substantial increases in travel times in a small area of the city and no increases in travel times in the rest of the city. Therefore, in order to obtain a true perspective on the full impact of any redeployment of fire companies, it is necessary to focus attention on smaller regions of the city. To this end, alarm boxes are grouped into the following five types of regions:

(1) Demand regions.
(2) Company response areas.
(3) The region affected by a change in deployment.
(4) Target hazards.
(5) The city as a whole.

In the following subsections, we define each type of region and discuss the usefulness of the information on each.

---

*See the definition of "First-Due Response Area" in Appendix A.

2.2.1. Demand Regions

To facilitate the generation and evaluation of deployment changes, a city should be partitioned into a number of demand regions before the deployment analysis is begun. A demand region is an area of the city that contains homogeneous fire hazards to life and property, potential fire-fighting problems, and alarm incidence. Population density, land use, housing stock, natural barriers, and alarm incidence should all be taken into consideration in determining the alarm boxes to be placed in each demand region. The alarm boxes constituting each of these regions remain fixed during the analysis. Taken together, the demand regions should constitute the entire city. The use of demand regions allows areas in the city having similar demand characteristics to be compared in terms of their fire protection measures. Also, by describing the characteristics of each region, those areas that, because of their greater fire hazards, require higher levels of fire protection can be identified. The analyst can then pay particular attention to these regions.

2.2.2. Company Response Areas

A company's first-due response area is defined as the set of alarm boxes to which it is the closest company. The size of this area (i.e., the number of alarm boxes) can change, depending on the nature of a particular deployment change. Because the size of this area is not constant it is not very useful for making travel time comparisons. However, it is useful for comparing workloads of individual companies under different configurations. Similar definitions hold for a company's second-due response area, third-due response area, etc. For a given deployment of companies the siting model automatically determines the response area for each company by estimating the travel distances between fire companies and alarm boxes (see Section 3.2). At the user's option, the program will list the alarm boxes constituting each response area.

2.2.3. Affected Regions

The siting model always considers two configurations of fire companies in performing its calculations and presenting its results—a configuration that is called "current" and one that is labeled "proposed."

The affected region is the set of alarm boxes to which the travel time for the first-due unit (i.e., first-due travel time) is different in the current and proposed configurations. By focusing strictly on the boxes in this area, the proposed configuration's effect on travel times to these boxes is not masked by averaging their travel times with those of boxes that would experience no change. Affected regions can also be defined (at the user's option) for second-due units, third-due units, etc. For each demand region and for the city as a whole, the siting model summarizes information on the alarm boxes whose travel times in the proposed configuration are improved, those whose travel times are degraded, and on all the affected boxes in the region. In addition, at the user's option it will list the alarm boxes that are included in the improved and degraded regions. The boxes included in the affected region are different for different configurations, unlike the boxes constituting the demand regions, which remain fixed.

2.2.4. Target Hazards

The siting model permits the user to identify some alarm boxes as more important than others for receiving a rapid response. These special locations are called target hazards and, of course, do not form a compact region. At the user's option, the siting model output provides information on travel times to each target hazard and, overall, to the set of target hazards.

2.2.5. The City As A Whole

We have mentioned that looking at travel time statistics averaged throughout the whole city can be very misleading. However, it is sometimes useful to look at this measure to get a quick idea of the overall effect of a deployment change. Additionally, it places the incremental benefit of each additional fire company in its proper perspective.

3. ASSUMPTIONS OF THE MODEL

A user of the siting model should be aware of the following five basic assumptions and relationships that are incorporated in the siting model so that he will understand the restrictions that they may impose on the results.

(1) Units are always available in their firehouses to respond to an incoming alarm.

(2) The closest units are always dispatched to an incident.

(3) Calculations are performed separately for each type of fire-fighting equipment.

(4) The travel distance between two locations can be estimated using one of two formulas.

(5) Travel time can be estimated from travel distance using an empirically determined function.

In the rest of this section we discuss each of these and examine the role that each plays in the model.

3.1. AVAILABILITY OF FIRE COMPANIES

The model assumes that, when a fire alarm is received, all of the fire-fighting units in the area of the incident are available to respond from their firehouses. This is a reasonable assumption for most cities, since the fire-fighting force is usually large compared to the alarm rate, implying that the probability is small that a specific company will be unavailable when an alarm is received from its response area. However, it is a poor assumption for a city such as New York, in which fire companies in some areas at some times are busy almost half the time. Even in cities in which fire companies spend less than five percent of their time responding to or working at alarms, there is always some probability that a company will be working at an incident when another alarm occurs in its response area. In these cases, the siting model's computed travel times will be underestimates of the actual travel times. If such situations occur frequently, a simulation model, such as the one described in [3], would be the most appropriate tool for evaluating alternative firehouse configurations.

3.2. DISPATCH OF CLOSEST UNIT

When calculating travel times and generating response area boundaries for a proposed firehouse configuration, the program assumes that the units geographically closest to an incident location will be the companies dispatched to an alarm there. This corresponds to the response policy in most cities. However, since the program calculates distances using an (x,y) grid of the city (see Section 3.4), it does not take into account the existence of physical barriers or other unusual geographic conditions. Therefore, some of the calculations may be underestimates of the actual travel distances. The analyst must take these considerations into account in using the results of the model.

The "closest unit" assumption is used to determine the response areas that result from a new configuration of fire companies. The method for determining response areas involves revising the alarm assignment lists in the model's data base. [For each alarm box the data base contains a list of companies of each type (e.g., engines, ladders) in increasing order of response distance (called an "alarm assignment" list) corresponding to the "current" configuration of fire companies.] If, for example, a modification to the current configuration involves adding a company, the program determines the travel distance between every alarm box and the location of the new company. Then, the new company is inserted in its correct place in the alarm assignment list for each box, based on its estimated distance to the alarm box.

3.3. CALCULATIONS FOR DIFFERENT TYPES OF EQUIPMENT

A configuration of fire companies usually includes several different types of fire-fighting apparatus (e.g., engine companies, which deliver water onto a fire, and ladder companies, which are responsible for rescuing people and ventilating a fire). Although there may be fire protection measures that require the simultaneous consideration of different types of companies (e.g., the travel time for the closest unit of any type to an alarm box), the siting model performs its calculations separately for each type of company. The version of the model documented in this report restricts its attention to two types of equipment (engines and ladders). If more than two types of companies are being studied, separate data bases can be constructed for every additional type or pair of types.

3.4. CALCULATING TRAVEL DISTANCES

There are many different ways of estimating the travel distance between two locations. For example, the distance can be measured on a map by following the actual route of response. Another method, which is more suited to the capabilities of a computer, involves assigning a pair of (x,y) coordinates to all locations of interest, such as firehouses and alarm boxes, within a city. Then the Euclidean (straight-line) distance between two points, specified by (x_1,y_1) and (x_2,y_2), can be calculated as:

$$D_E = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}.$$

Similarly, the right-angle distance between the two points is given by

$$D_R = | x_2 - x_1 | + | y_2 - y_1 | .$$

In cities where the streets follow a rectangular grid pattern, the travel distance between any two points can be estimated directly by D_R. However, in cities where streets do not follow such a pattern, we have found that a good estimate of the travel distance is obtained by multiplying the Euclidean distance by some factor, k, which is dependent on the street patterns and geography of the city (i.e., $D = kD_E$). Based on empirical data gathered in several cities, we have found that the value of k varies only slightly from city to city, with a value of $k = 1.15$ producing good estimates for most cities. The siting model permits the user to estimate travel distances by D_R or D. If distances are to be estimated using D, the user can supply the approximate constant, k. If none is specified, $k = 1.15$ is assumed.

3.5. ESTIMATING TRAVEL TIME FROM TRAVEL DISTANCE

Most of the fire protection measures that were discussed in Section 2 require the estimation of travel times. Yet, in most cities little information is available on how quickly fire companies respond to fires and how travel times vary with distance and time of day.

In order to obtain this information it is often necessary for a city to conduct a field experiment. The experiment involves the collection of data on responses made by selected units. For each response, the data collected include the distance traveled, the duration of the trip, the time of day, the location to which the unit responded, and information on weather and traffic conditions.

A computer program has been written at The New York City—Rand Institute that analyzes data collected in such an experiment to determine the response characteristics required as input parameters for the siting model. This program, as well as the mechanics of the experiment for collecting the data, are described in [9]. The results of a travel time experiment conducted by the New York City Fire Department appear in [10].

As a result of conducting the experiment in several cities, we have found that a single function usually provides a good representation of the relationship between travel time and travel distance at all times of the day and in all areas of a city. The functional form that we have found to be most useful provides a square-root relationship between time and distance up to some distance, d, and a linear relationship for distances greater than d. At the point d, the two functions intersect and have the same slope. This function is of the form

$$T(D) = \begin{cases} c\sqrt{D} & D \leq d \\ a + bD & D > d \, , \end{cases}$$

where T is the travel time and D is the travel distance. The parameters a, b, c, and d are determined from the results of the experiment and must be supplied to the siting model by the user.

Our experience in a wide range of cities has shown that the values of these parameters exhibit little variation from city to city. In performing an evaluation of alternative firehouse configurations it is really not necessary to obtain precise estimates of travel times, since relative times (i.e., which configuration produces *better* travel times) are more important than absolute times. Therefore, at least until a travel time experiment can be set up, performed, and analyzed, a city can obtain useful travel time estimates by using the following function, which is based upon experimental results in Trenton, New Jersey; Denver, Colorado; Wilmington, Delaware; and Yonkers, New York:

$$T(D) = \begin{cases} 2.1\sqrt{D} & D \leq .38 \text{ mile} \\ .65 + 1.7\,D & D > .38 \text{ mile,} \end{cases}$$

where D is the travel distance in miles and T is the travel time in minutes. If the user does not specify any value for a, b, c, and d, the program will automatically use the above function for estimating travel times.

4. OUTPUT FROM THE MODEL

4.1. PRINTED OUTPUT

The Firehouse Site Evaluation Model can produce output reports presenting results for each of the five types of regions described in Section 2: citywide, demand regions, special target hazards, the affected region, and company response areas. Separate reports are produced for each type of apparatus (engines and ladders) and for each response level (i.e., first-due, second-due, etc.) within a given apparatus type. The user can select the specific combination of output reports that he desires, either explicitly or by default, by use of the Output command (see Section 5). To facilitate comparisons, results for the current and proposed configurations are printed side by side. Statistics on the current configuration are stored in the model's data base and are not recomputed each time the user requests output. Statistics for the proposed configuration are computed after user-specified configuration changes have been made. Sample output reports produced by the program are shown in this section, and the output from a typical use of the program at a computer terminal is contained in Appendix B. The commands used to obtain the various output reports are described in Section 5.2.6.

The performance measures for all regions are computed using the same formulae. The calculations differ only in the way sets of alarm boxes are aggregated. In this section we will describe the calculations in general terms. A mathematical description of these calculations is provided in Appendix F.

The output reports for all regions except company response areas include the values (under the current and proposed configurations) of the following five measures for each response level and company type for which the user has requested output (names in parentheses are the abbreviations used to identify the measures on the output reports):

(1) Average travel time (AV.TR.T.): the sum of the travel times to each of the boxes in the region divided by the number of boxes. This is an estimate of the expected travel time to a randomly chosen point in the region.

(2) Average travel distance (AV.TR.D.): the sum of the travel distances to each of the boxes in the region divided by the number

of boxes.

(3) Average weighted travel time (AV.TR.T. TO ALARMS or AV.TR.T. TO STRUCTURALS): the product of the travel time to a box and the total number of alarms or structural fires (user option) that occurred in the last year at the box, summed over all boxes in the region, and divided by the sum of all the alarms or structural fires. This is an estimate of the expected travel time to a randomly-chosen alarm or structural fire occurring in the region.

(4) Average weighted travel distance (AV.TR.D. TO ALARMS or AV.TR.D. TO STRUCTURALS): the same as measure (3), but for travel distances instead of travel times.

(5) Maximum travel time (MAX TR.T.): the largest of the travel times for any of the boxes in the region.

The output for company response areas includes only two of these five measures—the average travel time [measure (1)] and the maximum travel time [measure (5)]—for each company's response area.

4.1.1. Citywide

This section includes a frequency distribution of travel times throughout the city. The frequency distribution gives the number of alarm boxes whose travel times fall within each of 31 half-minute intervals, ranging from 0 to 15 minutes. All travel times greater than 15 minutes are assigned to the last interval. The frequency distribution is particularly useful for studying the occurrence of long travel times under different fire company configurations.

Example: Citywide output for first-due engines:

```
1ST DUE ENGINE RESPONSE

CITYWIDE  (  74 BOXES) CURRENT  PROPOSED (  390 ALARMS,   95 STRUCTS.)
AV. TR.T.                1.59     1.84
AV. TR.D.                0.62     0.76
AV. TR.T. TO STRUCTURALS 1.57     2.06
AV. TR.D. TO STRUCTURALS 0.61     0.89
MAX TR.T.                3.38     3.91

DISTRIBUTION OF TRAVEL TIMES TO BOXES - CITYWIDE

TR TIME  0.5  1.0  1.5  2.0  2.5  3.0  3.5  4.0
# BOXES     13   22   14   10    3    4    8
```

In this example, there are 74 boxes in this (small) city. In the base period (the period of time represented by the incidence data in the data base) there were 390 alarms in the city, 95 of which were for structural fires. The current configuration of engine companies is being compared to a proposed configuration. In the current configuration the average first-due engine travel time, averaged over all boxes, is 1.59 minutes; in the proposed configuration the average would be 1.84 minutes. The average first-engine travel distance is .62 mile and would be .76 mile in the proposed configuration. The average travel time to structural fires would be expected to increase from 1.57 minutes to 2.06 minutes if the proposed configuration were implemented; the average travel distance to structural fires would be expected to increase from .61 mile to .89 mile; and the travel time to the alarm box located furthest from an engine company would increase from 3.38 minutes in the current configuration to 3.91 minutes in the proposed configuration (the identity of that alarm box might be different in each configuration).

The distribution of first-due engine travel times throughout the city shows that in the proposed configuration 13 boxes would have travel times between .5 and 1.0 minute, 49 would have travel times of less than 2.0 minutes, and 8 would have travel times greater than 3.5 minutes.

4.1.2. Demand Regions

Alarm boxes are grouped into demand regions by assigning them demand region numbers in the data base. The demand region output has precisely the same format as citywide output and includes frequency distributions of travel times to alarm boxes in each region. For an explanation of the output see Section 4.1.1.

Example: Region 5 output for first-due engines:

```
REGION  5  ( 121 BOXES) CURRENT  PROPOSED ( 1599 ALARMS,  279 STRUCTS.)
AV. TR.T.                 1.75      1.63
AV. TR.D.                 0.47      0.41
AV. TR.T. TO STRUCTURALS  1.65      1.63
AV. TR.D. TO STRUCTURALS  0.41      0.40
MAX TR.T.                 3.31      3.31


DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  5

TR TIME  0.0  0.5  1.0  1.5  2.0  2.5  3.0  3.5
# BOXES     1    9   36   47   27    0    1
```

4.1.3. Target Hazards

These do not form a compact region, but represent a region composed of alarm boxes that may be distributed throughout the city. The target hazard output includes a list of all target hazard locations, with the estimated travel time and travel distance to each location under the current and proposed fire company configurations. Those target locations whose travel time under the proposed configuration is different from the time under the current configuration are identified by asterisks.

Example: Target hazard output for first-due engines in a city with two target hazards.

```
 1ST DUE ENGINE RESPONSE

 2 TARGET HAZARDS        CURRENT  PROPOSED (   31 ALARMS,    4 STRUCTS.)
AV. TR.T.                  1.67      2.90
AV. TR.D.                  0.66      1.38
AV. TR.T. TO STRUCTURALS   1.67      2.90
AV. TR.D. TO STRUCTURALS   0.66      1.38
MAX TR.T.                  1.90      3.90


    BOX     TRV. DIST.    TRV. TIME
          CURR. PROP.   CURR. PROP.

    2113   0.79  0.79    1.90  1.90

*   2324   0.53  1.96    1.44  3.90
```

In this example, the fire department has identified two target hazards (alarm boxes 2113 and 2324). They experienced 31 alarms in the base period, of which 4 were structural fires. The changes in the proposed configuration do not affect the first-due engine travel distance or travel time to box 2113, but lead to a significant increase in the travel distance and travel time to box 2324.

4.1.4. Affected Regions

When any change is made in the arrangement of fire companies in the city, the travel distances and travel times to some alarm boxes will be changed. There will be changes in first-due responses to one set of boxes, changes in second-due responses to a different set of boxes, etc. If a company is added, all of the affected boxes will receive improved travel distances and times. If a company is deleted, all of the affected boxes will have their travel distances and times degraded. However, if a company is moved, the travel distances and times will be improved to some boxes and degraded to others. We therefore divide the affected region for each demand region, response level, and company type into two separate regions: (1) the set of alarm boxes receiving improved fire protection; and (2) the set of alarm boxes whose fire protection is degraded.

The affected region output consists of the five summary statistics for each of the two affected subregions and for the whole affected region. These are repeated for the city as a whole and for each demand region. At the user's option, lists of the specific boxes that have received improved and degraded fire protection can be printed.

Example: Affected region output for first-due engines:

```
1ST DUE ENGINE RESPONSE

CITYWIDE

 61 DEGRADED BOXES      CURRENT  PROPOSED (  445 ALARMS,    39 STRUCTS.)
AV. TR.T.                  1.05     1.41
AV. TR.D.                  0.32     0.51
AV. TR.T. TO STRUCTURALS   1.04     1.41
AV. TR.D. TO STRUCTURALS   0.32     0.51
MAX TR.T.                  1.86     1.97

252 IMPROVED BOXES      CURRENT  PROPOSED ( 1061 ALARMS,   241 STRUCTS.)
AV. TR.T.                  2.05     1.43
AV. TR.D.                  0.89     0.53
AV. TR.T. TO STRUCTURALS   1.87     1.22
AV. TR.D. TO STRUCTURALS   0.78     0.41
MAX TR.T.                  4.32     3.57

313 AFFECTED BOXES      CURRENT  PROPOSED ( 1506 ALARMS,   330 STRUCTS.)
AV. TR.T.                  1.86     1.43
AV. TR.D.                  0.78     0.53
AV. TR.T. TO STRUCTURALS   1.64     1.27
AV. TR.D. TO STRUCTURALS   0.65     0.44
MAX TR.T.                  4.32     3.57
```

```
REGION  1

  27 DEGRADED BOXES       CURRENT  PROPOSED (  240 ALARMS,    42 STRUCTS.)
AV. TR.T.                   1.01     1.41
AV. TR.D.                   0.29     0.51
AV. TR.T. TO STRUCTURALS    0.83     1.35
AV. TR.D. TO STRUCTURALS    0.21     0.48
MAX TR.T.                   1.49     1.77

  83 IMPROVED BOXES       CURRENT  PROPOSED (  317 ALARMS,    61 STRUCTS.)
AV. TR.T.                   2.29     1.73
AV. TR.D.                   1.02     0.70
AV. TR.T. TO STRUCTURALS    1.78     1.36
AV. TR.D. TO STRUCTURALS    0.73     0.49
MAX TR.T.                   4.32     3.57

 110 AFFECTED BOXES       CURRENT  PROPOSED (  557 ALARMS,   103 STRUCTS.)
AV. TR.T.                   1.97     1.65
AV. TR.D.                   0.84     0.66
AV. TR.T. TO STRUCTURALS    1.40     1.36
AV. TR.D. TO STRUCTURALS    0.52     0.48
MAX TR.T.                   4.32     3.57
```

In this example, the proposed change in the configuration of engine companies changes the first-due engine travel distance and time to 313 alarm boxes throughout the city. The travel time and distance are lengthened to 61 boxes (the average travel time to these boxes is increased from 1.05 minutes to 1.41 minutes). They are shortened to 252 boxes (resulting in the average travel time to these boxes decreasing from 2.05 minutes to 1.43 minutes). Averaged over all the affected boxes, the average travel time is reduced from 1.86 minutes to 1.43 minutes. The average travel time to structural fires that are expected to occur at these boxes is decreased from 1.64 minutes to 1.27 minutes.

The same pattern of output is repeated for the affected boxes in each demand region. For the sake of brevity, only affected region output for Demand Region 1 is shown.

4.1.5. Company Response Areas

The program determines company response areas from the alarm assignment lists for each box. Current response areas are determined from the alarm lists in the data base. Proposed response areas are determined from these lists after user-specified configuration changes have been applied. An alarm box will be assigned to the jth-due response area of company i if the company is the jth company in its alarm assignment list; i.e., if there are j-1 closer companies to the box.

For each company's response area, the program calculates and prints the number of boxes constituting the area, the (unweighted) average and maximum travel times in the area, and the number of alarms and structural fires that occurred in the area during the past year. If the company's response area in the proposed configuration is different from its current response area, it is flagged with an asterisk in the output listing.

At the user's option, the program will print lists of the boxes that constitute each company's response area.

Example: First-due response areas for engine companies:

1ST DUE ENGINE RESPONSE

	CO.	BOXES		AV. TR.T.		MAX TR.T.		ALARMS		STRUCTURALS	
		CURR.	PROP.	CURR.	PROP.	CURR.	PROP.	CURR.	PROP.	CURR.	PROP.
*	1	32	40	2.15	2.46	3.38	3.91	154	236	41	61
	2	18	18	1.09	1.09	1.36	1.36	56	56	14	14
*	5	2	0	1.53	0.00	1.90	0.00	4	0	0	0
*	8	8	0	1.45	0.00	1.65	0.00	82	0	20	0
*	10	14	16	1.06	1.13	1.33	2.39	94	98	20	20

The sample output above shows first-due company response area statistics for the five engine companies in this hypothetical city. The response area for Engine 1 is changed in the proposed configuration. This is indicated by an asterisk at the left of the output line for that company. The number of boxes in Engine 1's first-due response area would increase from 32 to 40; the average travel time to all boxes in its first-due response area would increase from 2.15 minutes to 2.46 minutes; and the travel time to the furthest box in its first-due area would increase from 3.38 minutes to 3.91 minutes. There were 154 alarms at the boxes in Engine 1's current first-due area during the base period. This would increase to 236 alarms in the proposed configuration. The number of structural fires that occurred during the base period in its first-due area would increase from 41 to 61.

Engine 2's first-due response area remains unchanged in the proposed configuration, while Engine 5 and Engine 8 have been eliminated (the command that accomplished this has not been shown). As a result, Engine 5 and Engine 8 have no boxes in their new first-due response areas, so all of their "proposed configuration" statistics are printed as zeros.

4.2. ALARM ASSIGNMENT OUTPUT

The siting model program has the capability of creating a computer file in printable format that contains the newly generated alarm assignment lists for each alarm box under the proposed configuration. This capability is triggered by an optional user specification described in Section 5.2.6. This option is designed to provide a fire department with an easy way of generating new response assignments for its fire companies when company locations are changed.

Example: Sample alarm assignment output:

```
129 ENGINES
      2    6   10    4    5    1    9    8    7   17
    LADDERS
      1    2   12    6   11

130 ENGINES
      4    5    2   10    1    6    9    7    8   17
    LADDERS
      6    1   12    2   11

131 ENGINES
      4    5    2   10    1    6    9    7    8   17
    LADDERS
      6    1   12    2   11

132 ENGINES
     10    2    4    5    1   20    9    6    8    7
    LADDERS
     12    1    6    5    2

133 ENGINES
     10    2    4    5    1   20    9    6    8    7
    LADDERS
     12    1    6    5    2

134 ENGINES
      2    1    6    4    5   10    9    7    8   17
    LADDERS
      1    2    6   12   11

135 ENGINES
      2    1    6    4    5   10    9    7    8   17
    LADDERS
      1    2    6   12   11
```

In this example alarm assignment lists are shown for 7 alarm boxes (alarm boxes numbered 129-135). Each list contains the identification numbers of the closest 10 engines and 5 ladders to the box. For example, Engine 2 and Ladder 1 are the first-due units at box 129, and Engine 4 and Ladder 6 are first-due at box 130.

4.3. NEW DATA BASE

Once the user has made a final decision concerning one or more changes in the configuration of fire companies, he may wish to make these changes permanent in the siting model data base. The program facilitates this by providing an option to generate a new data base from the proposed configuration. The user can then specify which of a number of data bases is to be used to define the "current" configuration.

These options are specified by using the "O" and "I" commands described in Section 5.2.

4.4. MAPPING OUTPUT

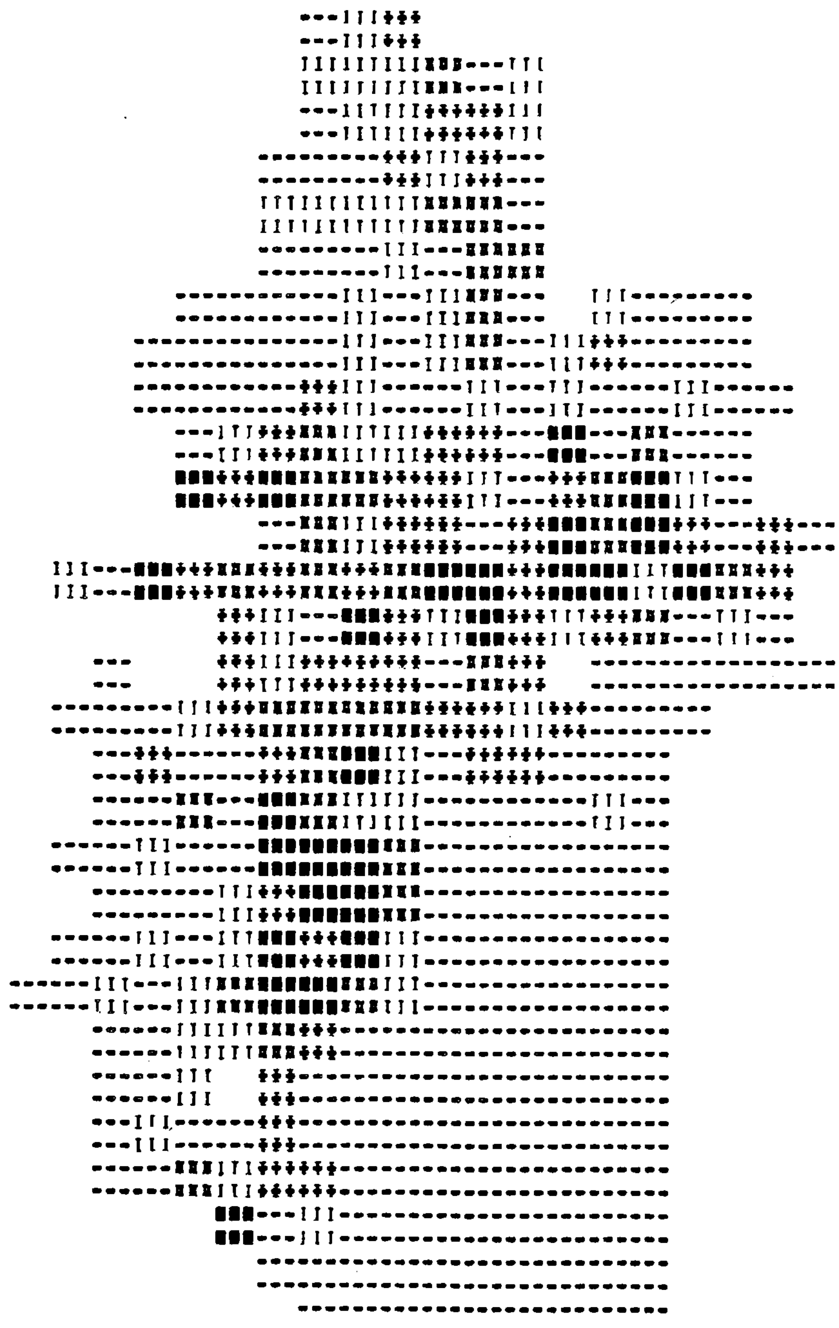
At the user's option, the siting model can produce a file consisting of one record for each alarm box, containing its (x,y) coordinates, total and structural alarm counts, engine and ladder travel times, and affected region indicators.* Using this file as input to the GRIDS computer-mapping system [11], the user can obtain shaded maps showing the geographic pattern of alarms, travel times in the city, and regions affected by a change in the arrangement of fire companies. This option is selected by the user specifying the "M" parameter in the "O" command (see Section 5.2.6). The format of the resulting file is given in Table C.3.** An example of the type of map that can be produced from this file using GRIDS is given in Fig. 1, which shows the 1973 geographic density of structural alarms in Jersey City, New Jersey. The alarm density is indicated by the shading on the map. For example, those areas of the city that are shaded with dashed lines experienced fewer than 3 structural fires per square mile in 1973.

Another example of the type of map that can be produced from this file using GRIDS is given in Fig. 2, which shows the areas of Trenton, New Jersey that would be affected by the engine company redeployment shown in Fig. 5. Periods in the map indicate areas whose first-due engine response times would be unchanged; plus signs indicate areas that would experience improved response times; and the darkest shading indicates areas with degraded response times.

---

*These indicators show whether a given travel time is improved, degraded, or remains the same in the proposed configuration.

**Tables identified by a letter may be found in the corresponding appendix at the end of this document.



LEGEND: NUMBERS UNDER EACH SYMBOL BLOCK REPRESENT
MINIMUM AND MAXIMUM STRUCTURAL ALARMS/SQUARE MILE

.0000000	3.000000	6.000000	11.00000	21.00000
3.000000	6.000000	11.00000	21.00000	OR OVER

Figure 1. Sample GRIDS output showing the geographic density of structural alarms in Jersey City, New Jersey.

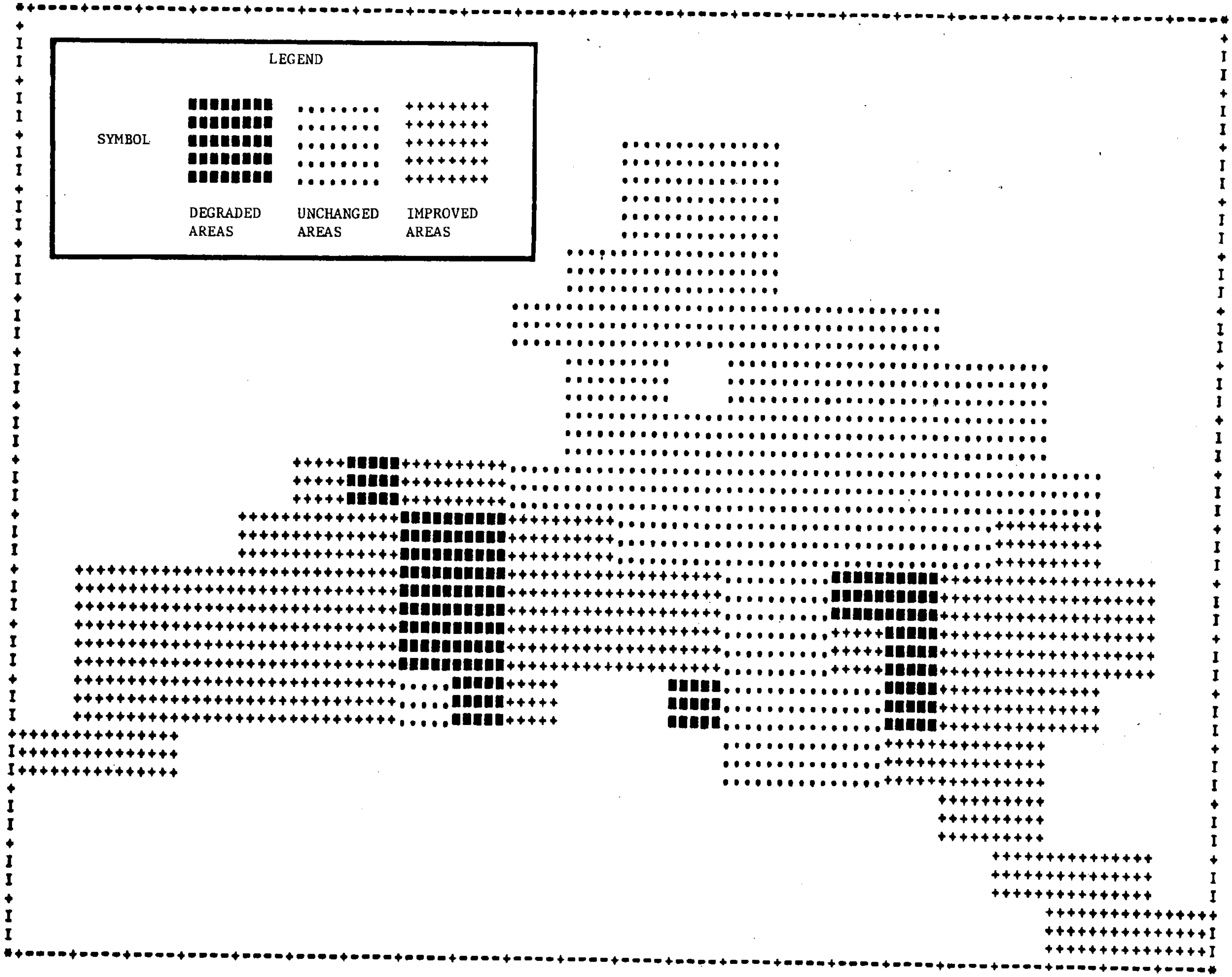


Figure 2. First-due engine affected areas resulting from proposed redeployment of engine companies in Trenton, New Jersey.

5. THE COMMAND LANGUAGE FOR THE SITING MODEL PROGRAM

The user controls the operation of the siting model program by typing a sequence of instructions, or commands, at a computer terminal. These commands specify proposed changes in the configuration of fire companies within the city and control the types of output produced by the program.

The user will generally start by typing a sequence of commands specifying changes to be made to the current configuration of companies. The commands might specify moving a company to a different location, adding a new company at a specific location, or removing a company from service. Following the sequence of commands that define a proposed configuration, the user can request that the program print output at his terminal that compares various statistics about the current and proposed configurations.

On examining the output, the user may decide that he wishes to evaluate the effects of additional changes to the proposed configuration. Since the program "saves" all changes entered unless otherwise instructed, only the additional changes need to be entered. This process of specifying and evaluating proposed configurations can be continued as long as desired.

Entering a command to change the configuration is like moving a piece in a board game such as chess. When output is requested, statistics are accumulated on the basis of the location of the companies on the "board." An important difference between the siting model program and most board games is that the program "remembers" the changes that have been made in the current configuration to produce the proposed configuration. This allows the user to "back up," eliminating a given number of changes, and to proceed from there.

There are two important points concerning the manner in which the program saves changes that should be noted here. The first is that changes are *not* saved between sessions on the computer system unless the user specifically saves a copy of the modified data base by using the "U" parameter of the Output command (see Section 5.2.6). The second is that there is a limit to the number of changes that can be saved at one time. The limit depends on how the program is installed on the computer system and usually will not concern the user. The program provides automatic notification if this limit is exceeded.

The siting model offers a large number of options to the user through the use of various commands. These options include:

- Adding, moving, and deleting fire companies to specify different configurations.
- Printing the list of changes that led from the current configuration to the proposed configuration.
- Returning to a prior proposed configuration.
- Selecting a variety of statistics to be printed out that compare the current and proposed configurations.
- Generating a new data base from the proposed configuration.
- Specifying the data base that is to be used to define the current configuration.
- Generating an alarm assignment file.
- Generating a data file that can be used for computer mapping.
- Terminating execution of the program.

The commands accepted by the program are described completely below. Appendix B illustrates the use of the commands by presenting an actual session at a computer terminal.

5.1. ENTERING COMMANDS

All commands described below can be entered only in response to the message "COMMAND?" typed by the program. The program does not impose any restrictions on the order in which commands are entered, but the order does affect the program's output.

All commands consist of a single letter (such as "A" to add a company, "M" to move a company, etc.) followed, in most cases, by additional information (parameters). When entering commands, the letter identifier must be entered first and at least one space must be entered between the identifier and any parameters.

Commands are terminated by a carriage return. This means that a maximum of one line of input can be used to specify any command.

5.2. COMMAND DESCRIPTIONS

The following conventions are used in the descriptions of the parameters of the commands:

- Braces ({ }) mean that one item out of the enclosed list of items must be entered.
- Brackets ([ ]) indicate that the enclosed item or items are optional and either or none of them may be entered.
- Three dots (. . .) are used to indicate that the preceding item(s) can be repeated any number of times in succession.
- Wherever ordinary parentheses ( ) are shown, they must be included in the typed command.
- Fields shown as nn, mmmm, etc., indicate that a decimal digit (0-9) should replace each of the letters when entering the command. Leading zeros prefixed to numbers entered are ignored and may be omitted. Thus, "E1" and "E01" both identify Engine Company No. 1. All other items in the format descriptions must be entered exactly as shown.
- An underlined item is the default value used by the program if the user makes no entry for a particular parameter.
- Alarm boxes must be assigned numerical designations consisting of no more than four digits.

5.2.1. A (Add a Company)

Format: A $\begin{Bmatrix} E \\ L \end{Bmatrix}$ - mmmm [, . . .]

Action: Adds an engine or ladder company (indicated by E or L) at box mmmm. Any number of companies (≥ 1) can be added with one command (subject to the one-line command length restriction). The program automatically assigns a unique identifier to each added company. These identifiers are displayed in the output of the "P" (Print) command (see below).

Example: A E-901, E-3216, L-3216

This command adds an engine company at box 901 and an engine and ladder company at box 3216 (see note *c* in Appendix B).

5.2.2. C (Cancel Change(s))

Format: C $\begin{Bmatrix} n \\ * \end{Bmatrix}$

Action: If "*" is entered, this command returns the proposed configuration of the city to the "current" (original) configuration. If "n" is entered, it returns the proposed configuration to the way it was before the last n change commands (Add, Move, or Delete) were entered. To list the changes that produce the proposed configuration (in the order entered), use the "P" command described below.

Example: C 2

This command returns the proposed configuration to the state it was in before the last two change commands were entered. (For additional information, see notes *e*, *t*, and *u* in Appendix B).

5.2.3. D (Delete a Company)

Format: D $\begin{Bmatrix} E \\ L \end{Bmatrix}$nn[, . . .]

Action: Removes engine or ladder company nn. Several companies can be removed using one Delete command.

Example: D E2, E10, L4

This command eliminates engine companies 2 and 10 and ladder company 4 from the configuration (see note *b* in Appendix B).

5.2.4. E (Exit)

Format: E

Action: Terminates execution of the siting model program and returns control to the operating system (see note *z* in Appendix B).

5.2.5. M (Move a Company)

Format: M $\begin{Bmatrix} E \\ L \end{Bmatrix}$nn - mmmm [, . . .]

Action: Moves engine or ladder company nn to box mmmm. Several companies can be moved using one Move command.

Example: M E10-2104, L2-905

This command moves engine company 10 to box 2104 and ladder company 2 to box 905 (see note *a* in Appendix B).

5.2.6. O (Output)

The "O" command (the letter O) is used to specify the set of output to be produced in a given run of the model. The user's entering of an O command is also a signal to the program to complete its calculations and produce the requested output. The command has eight parameters associated with it, each of which has a different effect on the output produced. The user may include any number of these parameters, separated by commas, with the O command. Default values are used for any parameters not specified. We discuss each of the parameters separately below. Several examples of typical O commands are shown following this discussion. Other examples are shown in Section 4 and in Appendix B. These examples are referenced in parentheses following the definition of the parameter.

5.2.6.1. Parameters of the O Command.

(a) $C = \left(\begin{Bmatrix} E \\ L \\ \underline{E,L} \end{Bmatrix}\right)$

The C parameter selects the company type(s) for which output is to be printed. C = (E) selects engine output only, C = (L) selects ladder output only, and C = (E,L) (the default value) selects both. (See note *v* in Appendix B.)

(b) $D = \begin{Bmatrix} n \\ \underline{1} \end{Bmatrix}$

The D parameter specifies the response level to which output will be printed, e.g., D = 3 will cause printing of first-due, second-due, and third-due output of the types selected by other parameters. (See note *f* in Appendix B.)

(c) R = ([C] [,D] [,A] [,T])

The R parameter specifies the regions for which statistics will be printed. The statistics for the regions are printed as modified by the parameters discussed in (a) and (b) above. The region specifications are:

- R = (C) causes printing of company response area statistics. (See Section 4.1.5 and note *m* in Appendix B.)
- R = (D) causes printing of demand region statistics. (See Section 4.1.2 and notes *i* and *k* in Appendix B.)
- R = (A) causes printing of statistics about boxes affected

by the change from the current to the proposed configuration. (See Section 4.1.4 and note *o* in Appendix B.)

- R = (T) causes printing of target hazard statistics. (See Section 4.1.3 and notes *q* and *r* in Appendix B.)

The R parameter can be entered with any number (including none) of the permissible arguments, separated by commas. It is important to note that, if the R parameter is entered, only output for the regions specifically requested will be printed. However, if the R parameter is not used at all, C, D, A, and T output statistics will be produced by default. A group of statistics relating to the city as a whole will always be printed for each type of company selected by the C parameter (described in (a) above), even if no arguments are included for the R parameter (see example (c) in Section 5.2.6.2).

(d) $L = \begin{Bmatrix} Y \\ \underline{N} \end{Bmatrix}$

The L parameter specifies whether or not lists of the boxes that constitute the company response areas and affected regions are to be printed (Y means yes; N means no). Even if L = Y is specified, a list will be printed for a region of a given type (C or A) only if it is also selected (either explicitly or by default) by the R parameter. (See Sections 4.1.4 and 4.1.5 and notes *v* and *y* in Appendix B.)

(e) $W = \begin{Bmatrix} \underline{S} \\ A \end{Bmatrix}$

The W parameter specifies the type of weighting to be used in calculating the summary statistics for a region (see Section 4.1). W = S causes average travel times and distances to structural fires to be printed (i.e., the travel times are weighted by the incidence of structural fires at alarm boxes). W = A causes average travel times and distances to all alarms to be printed.

(f) $M = \begin{Bmatrix} Y \\ \underline{N} \end{Bmatrix}$

The M parameter specifies whether or not the program is to write a file of data on the proposed configuration that can be used to produce shaded maps showing the geographic distribution of alarms, average travel times, and the extent of the affected region (Y means yes; N means no). (See Section 4.4.) Figs. 1 and 2 in Section 4.4 are examples of the types of maps that can be produced using this file. See Table C.3 and Table D.2 for a description of the file that is created if the parameter M = Y is specified.

(g) $A = \left\{\begin{matrix} Y \\ \underline{N} \end{matrix}\right\}$

The A parameter specifies whether or not the program is to create a file containing alarm assignment information (see "Alarm Assignment List" in Appendix A) for each box in the proposed configuration (Y means yes; N means no). The resulting file can subsequently be printed. It facilitates the updating of alarm assignment lists for boxes after a decision has been made to implement a particular configuration of fire companies. An example of this output is given in Section 4.2. See Table D.2 for a description of the device requirements of the file that is created if A = Y is specified.

(h) $U = \left\{\begin{matrix} nn \\ \underline{0} \end{matrix}\right\}$

The U parameter specifies whether or not the program will create a new data base on the file specified by FORTRAN file reference number nn (see Section 4.3). The value of nn must not exceed 99. A value of 0 (the number zero) indicates that no new data base will be written. If a new data base is written, it will describe the proposed configuration of fire companies. This allows the user to save the data base corresponding to a given configuration. The value of nn must not specify any of the other files being used by the program (see Table D.2). See the discussion of the "I" command below for a description of how to reference the newly-created data base.

5.2.6.2. Examples of the Use of the O Command.

(a) O

This command causes the program to produce the output specified by the default values for all parameters. First-due engine and ladder output for all types of regions will be printed. No box list, alarm assignment, or mapping output will be produced. Travel times and distances will be weighted by structural fires.

(b) 0 D = 2, C = (E), R = (C,D,T), L = Y

This command causes the program to print first- and second-due engine company output [D = 2, C = (E)] for company response areas, demand, and target hazard regions, and for the city as a whole [R = (C,D,T)]. A list of the boxes in the first- and second-due response areas of each engine company is printed [L = Y]. The program will not produce any affected region output for engines, nor will there be any output for ladder companies.

(c) 0 R = ( ), D = 3, M = Y, A = Y, U = 25

This command causes the program to print first-, second-, and third-due [D = 3] engine and ladder output [the default for the C parameter] for the city as a whole only [R = ( )]. The program will create a file with one record for each box containing its identifier, (x,y) coordinates, total and structural alarm counts, first-, second-, and third-due engine and ladder travel times, and affected region indicators [M = Y]. This file can be used to to create shaded maps using the GRIDS system (see Figs. 1 and 2 in Section 4.4). The program will also create a printable file containing alarm assignment information for each box [A = Y] and a new data base file on FORTRAN file reference unit number 25 [U = 25].

5.2.7. I (Change the Input File)

Format: I U = nn

Action: Specifies the FORTRAN file reference number of the file containing the data base that is to be used during this and subsequent runs to define the "current" configuration of fire companies. The value of nn must not exceed 99. A value of 0 indicates that the default file reference number will be used (see Section 7). Otherwise, nn must specify a file residing on a device such as a magnetic tape drive, disk, drum, etc., that supports binary files and can be rewound. The I command can be used in conjunction with the U = nn option of the 0 command. This allows the user to specify that a siting model data base be written on FORTRAN unit nn using data for a proposed configuration, and that this new data base be used to define the "current" configuration in subsequent runs. The designation of unit nn for the input data base remains in effect until another I command is issued or the user terminates the session.

Example: I U = 25

This command causes the file assigned to FORTRAN unit 25 to be used as the siting model data base. The file must have been created with the DATGEN program or the U = nn option of the O command in the SITE program.

5.2.8. P (Print Changes)

Format: P

Action: Causes the siting model to print a list of the changes in the arrangement of fire companies that will produce the "proposed configuration." The changes are associated with the input commands used to enter them, which are numbered in the order they were entered. (This is also the order in which they will be applied.) When several changes have been made using a single command, one line is printed for each change. (See notes *a* - *e* in Appendix B.)

5.3. COMMAND ERRORS

The program responds to errors in commands by printing a message that identifies the type of error. If the program indicates that a command is in error, it treats the command as if it had not been entered and takes no action on it.

6. AN EXAMPLE OF THE USE OF THE SITING MODEL: REDEPLOYING FIRE COMPANIES IN TRENTON, NEW JERSEY

We will use a specific example, the reallocation of fire companies in the city of Trenton, New Jersey, to illustrate how the Firehouse Site Evaluation Model can be used in developing and evaluating new deployment policies for a city. All of the firehouses in Trenton were old, and some were in serious need of renovation or reconstruction. This provided an opportunity for the city to undertake a comprehensive examination of the deployment of all of its fire companies to determine if their locations could be improved. We will begin by characterizing the demand for fire-fighting service in Trenton. We will then show how the siting model was used to find better locations for Trenton's nine engine and four truck (or ladder) companies. A complete description of the Trenton study is presented in [3].

6.1. DEMAND REGION CHARACTERISTICS

In order to facilitate the analysis of the deployment of fire companies in Trenton, the city was partitioned into six demand regions. The basis of the division was the perceived homogeneity of fire incidence, fire hazards, and potential fire-fighting problems within each region. Population density, land use, housing stock, and alarm rates were all taken into consideration in determining these boundaries. Figure 3 is a map of Trenton, showing the boundaries of the six demand regions and the locations of all fire-fighting equipment deployed in the city.

Table 1 summarizes some of the demographic and alarm incidence characteristics of the six regions. Demand Region 4 is the most densely populated part of Trenton. This region also encompasses the central business district, as well as the state, county, and municipal office buildings, so that in the daytime the population is swelled by shoppers and businessmen. These factors result in a fire hazard in Region 4 that is greater than in other regions, as well as a higher alarm density, particularly the density of structural fires. Demand Region 2 ranks second in terms of both population density and alarm density. It is characterized by relative population stability and mixed land use: industrial and residential. Demand Region 6 is ranked third in terms of population density and Region 5 ranks third in alarm density. The housing

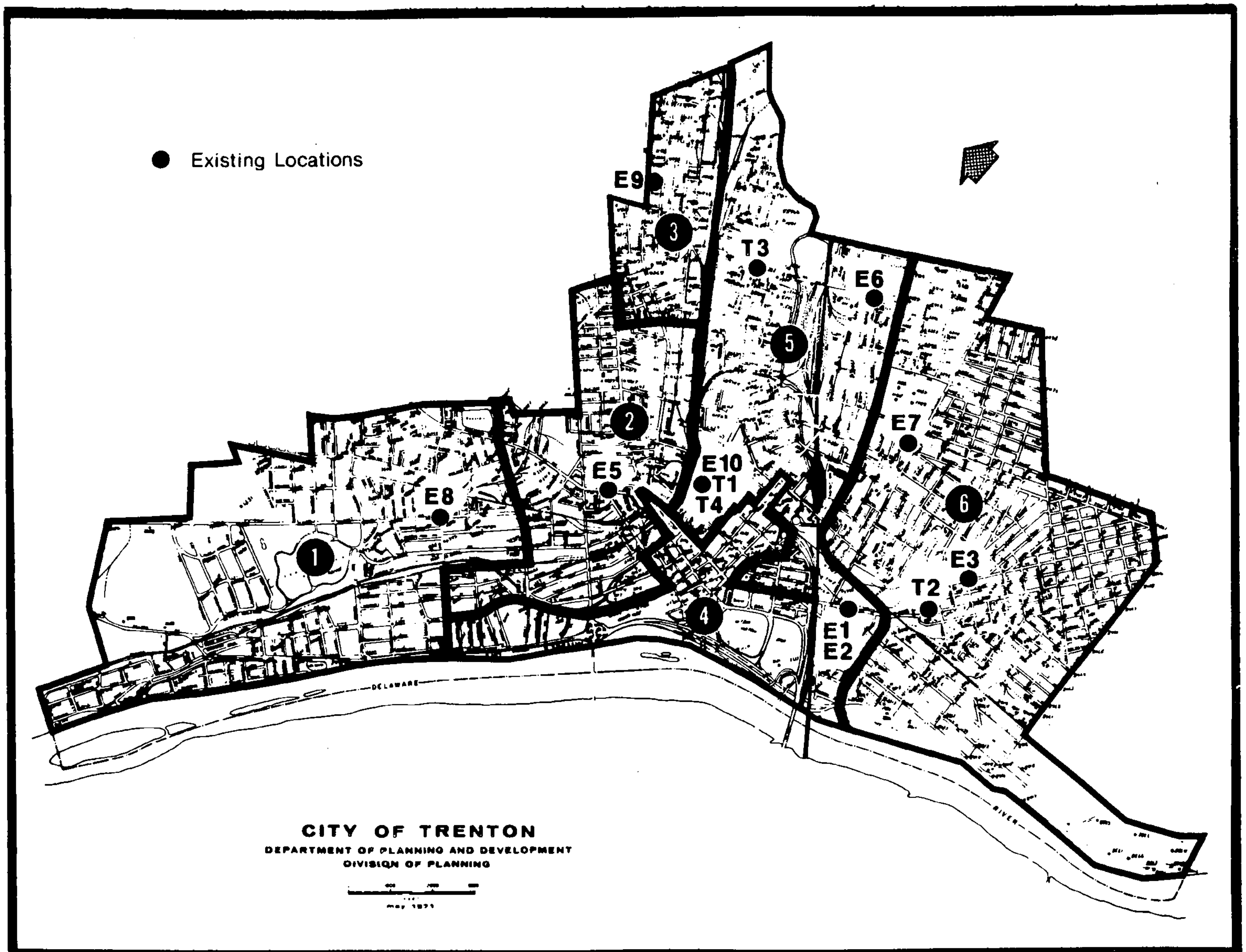


Figure 3. Trenton: Map of demand regions and fire company locations.

Table 1

SUMMARY OF DEMAND REGION CHARACTERISTICS

Demand Region	Area (sq. miles)	Population				Alarm Incidence, 1973					
						No. of Alarms			Alarm Densities (alarms per sq. mile)		
		1960	1970	% Change	1970 Population Density (pop./sq.mile)	Total	Struc-tural	False	Total	Struc-tural	False
1	1.76	18,017	15,753	-12.8%	8,951	609	111	216	346	63	123
2	.92	18,147	17,408	- 4.1%	18,913	676	176	189	735	191	205
3	.48	6,700	6,015	-10.2%	12,521	139	29	15	290	60	31
4	.38	12,753	10,744	-15.7%	28,274	379	94	84	997	247	221
5	2.72	19,182	18,240	- 4.9%	6,706	1,123	323	268	413	119	99
6	2.20	39,370	36,278	- 7.8%	16,486	724	162	156	329	74	71
Totals	8.46	114,169	104,438	- 8.5%	12,345[a]	3,650	895	928	431[a]	106[a]	110[a]

[a]These figures are average densities, not totals.

in these regions consists mostly of moderate density, single-family residences and row housing of somewhat higher density. Demand Regions 1, 3, 5, and 6 are all similar to each other; they have similar housing stock and relatively low population and alarm densities. Whatever measures of fire protection are used, a reasonable allocation of resources within the city would be one that produces about the same average travel times in each of these four regions.

6.2. THE CURRENT DEPLOYMENT OF FIRE COMPANIES

6.2.1. First-Due Engines

Table 2 shows first- and second-due travel times obtained from the siting model for the current configuration of engines and trucks in Trenton. The results show, for example, that the first-due engine travel times in the demand regions are not consistent with the relative hazards in the regions. Region 4, Trenton's downtown area, should have enough engine companies assigned in and around it to produce the lowest average travel times in the city, since it has the greatest hazards, the highest population density, and the highest alarm density. However, Region 4 has the fifth highest average first-due engine travel time of the six demand regions. Therefore, in order to match the supply of fire companies more closely to the demand, any redeployment of fire companies should try to reduce first-due engine travel time in Region 4 relative to the other demand regions.

Current maximum first-due engine travel times exhibit considerable regional variation, ranging from 1.85 minutes in Region 3 to 5.01 minutes in Region 1. Region 6 has the second highest maximum first-due engine travel time—4.17 minutes—followed by Region 4, with a surprisingly high maximum first-engine travel time of 3.38 minutes. The travel time frequency distributions obtained from the siting model show that, of the 664 fire alarm boxes located in Trenton, only 36 are located so that the closest engine requires more than 3 minutes in order to travel to them. Eighteen of these 36 boxes are located in Region 1, eight in Region 6, five in Region 4, and five in Region 2. This information indicates that a redeployment of fire companies should seek to reduce the number of such long

Table 2

ESTIMATED CURRENT FIRST- AND SECOND-DUE TRAVEL TIMES[a]

Demand Region	First-Due Engine		Second-Due Engine		First-Due Truck		Second-Due Truck	
	Average Travel Time	Maximum Travel Time	Average Travel Time	Maximum Travel Time	Average Travel Time	Maximum Travel Time	Average Travel Time	Maximum Travel Time
1	2.00	5.01	3.82	7.50	4.02	6.52	4.02	6.52
2	1.41	3.35	2.28	4.24	1.75	3.25	1.80	3.25
3	1.15	1.85	2.79	4.03	1.74	2.34	2.93	4.05
4	1.62	3.38	1.68	3.91	1.65	3.02	1.77	3.02
5	1.19	1.80	1.82	2.69	1.35	2.66	2.11	3.97
6	1.53	4.17	1.91	3.63	1.68	3.60	3.28	5.79
Citywide	1.51	5.01	2.31	7.50	2.02	6.52	2.78	6.52

[a] All travel times in minutes.

response times, particularly for responses in Region 4.

6.2.2. Second-Due Engines

The citywide average second-due engine travel time is 2.31 minutes. By region, second-due average travel times for engines range from 1.68 minutes in Region 4 to 3.82 minutes in Region 1, and are more consistent with fire protection demands than are the first-due engine travel times. Region 4, which is surrounded by five engine companies, has the lowest average second-due travel time, receiving protection that, in this case, reflects the fire-fighting hazards in the central business district, where more units must arrive at the scene of a fire quickly. The less populated, more residential areas, such as Regions 1 and 3, have higher second-engine response times, consistent with their lower hazards and alarm rates.

The maximum second-due engine travel time in Region 1 is 7.50 minutes. This is more than 3 minutes longer than the maximum second-due travel time in any other region of the city. Except for Region 1, maximum response times are not unreasonably high, although a redeployment might be able to improve them somewhat. Region 4, which has the greatest hazards, and Region 2, which ranks second in terms of hazard, rank third and fifth, respectively, in terms of maximum second-due travel time. Of the 53 fire alarm boxes to which second-engine travel time exceeds four minutes, 49 are located in Region 1, 2 in Region 2, and 2 in Region 3, emphasizing the magnitude of the second-due engine problem in Region 1.

6.2.3. First-Due Trucks

An examination of current truck travel times does not reveal as many problems as were found with engines. The only real difficulty arises in Region 1, an area that is served by a truck company that was moved from its original house because the house could no longer be used. However, no change was made in its area of responsibility. Therefore, the first-due truck travel times to boxes in Region 1 are very high, averaging 4.02 minutes, compared to uniformly well-balanced average travel times ranging from 1.35 minutes to 1.74 minutes in the other five regions.

Maximum first-due truck travel times are also well balanced. The exception again is Region 1, which has a maximum travel time of 6.52 minutes. There are 50 alarm boxes to which first-due truck travel time exceeds four minutes and all 50 are located in Region 1.

In looking for alternative locations for the four truck companies, an appropriate strategy would be to reduce the average and maximum first-due truck travel times in Region 1 and to make the first-due truck travel times in Regions 2 and 4 better than the other regions, because of the higher hazards in those regions.

6.3. REDEPLOYING CURRENT FIRE-FIGHTING RESOURCES

In this section, we show how the siting model was used to evaluate alternative redeployments of Trenton's existing fire companies to find ways to improve fire protection levels and make them more consistent with the regional fire hazards and demands. The siting model was used to examine several redeployment configurations suggested both by the Trenton Fire Department and by the Institute's analysis. The results presented here are for those configurations that were selected by the city as the best, based on a number of factors besides the numbers produced by the siting model.

6.3.1. Trucks

In analyzing the deployment of the four truck companies in Trenton, we concentrated on first-due truck times for the reasons stated above. We present results for two alternative configurations, named Option 1 and Option 2, and a method for comparing them. In each configuration, Truck 1 is moved to a new location west of its current location. Under Option 1 it is moved to point A on the map in Fig. 4. Under Option 2 it is moved to point B.

Table 3 summarizes the effects on travel time in the various demand regions resulting from the two redeployments of truck companies. Under Option 1, average citywide travel time drops from 2.02 minutes to 1.75 minutes, a decrease of 13 percent. Average travel times in the six demand regions range from 1.27 minutes in Region 2 to 3.00 minutes in Region 1. Region 4, the central business district, has an average travel time of 1.40 minutes. Thus, the average travel times in the high-hazard regions are still substantially lower than the average travel time in Region 1.

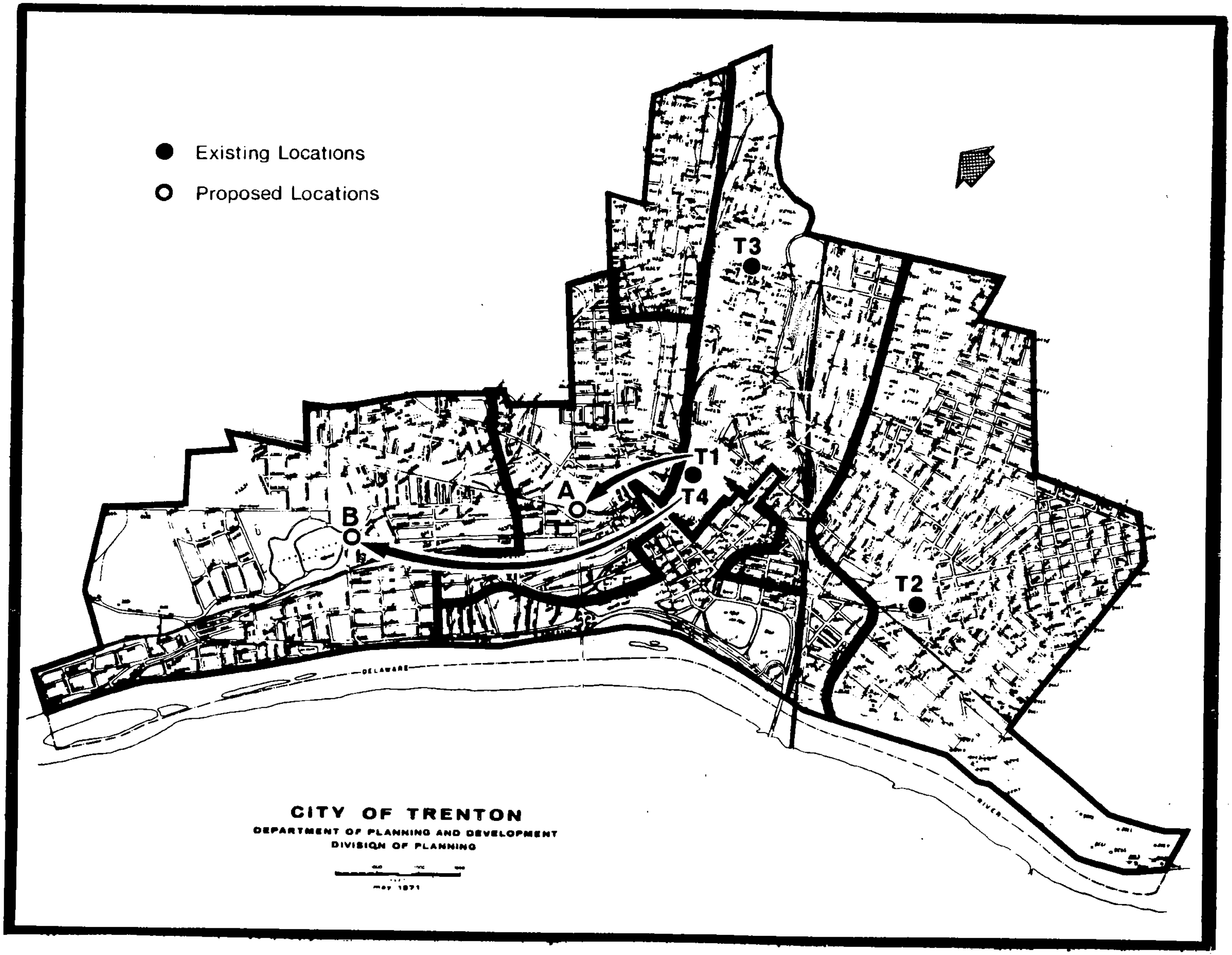


Figure 4. Proposed redeployment of truck companies.

Under Option 2, the average citywide travel time is further reduced by 21 percent. The average travel times in the six demand regions show less variability, ranging from 1.35 minutes in Region 5 to 1.74 minutes in Region 3. In fact, the expected travel time for the first-arriving truck is almost the same in each of the regions. But the average and maximum travel times in Regions 2 and 4 are higher than under Option 1.

Table 3

AVERAGE AND MAXIMUM FIRST-DUE TRUCK TRAVEL TIMES[a]

Area	Current		Option 1		Option 2	
	Average	Maximum	Average	Maximum	Average	Maximum
Citywide	2.02	6.52	1.75	5.41	1.60	3.60
Region 1	4.02	6.52	3.00	5.41	1.67	3.57
Region 2	1.75	3.25	1.27	2.18	1.66	2.25
Region 3	1.74	2.34	1.74	2.34	1.74	2.34
Region 4	1.65	3.02	1.40	2.36	1.54	2.94
Region 5	1.35	2.66	1.35	2.60	1.35	2.66
Region 6	1.68	3.60	1.68	3.60	1.68	3.60

[a]All travel times in minutes.

Both of these options offer better fire protection than the current truck deployment, since most average and maximum travel times are reduced and none are increased. But the question of which of the two options is better is a far more difficult one to answer. In attempting to answer this question, we look at the affected region* for each of the options and then consider three aspects of fire protection:

(1) travel time to structural fires;

(2) coverage; and

(3) hazards.

Useful information for performing this analysis, all of it obtained from the output of the siting model, is presented in Table 4.

*Recall that the affected region is the set of alarm boxes to which travel time has changed as a result of the redeployment.

Table 4

SUMMARY OF CHANGES IN THE FIRST-DUE AFFECTED REGION FOR EACH OPTION

Affected Region Measure	Option 1	Option 2
Reduction in average travel time (mins.)	0.90	1.97
No. of alarm boxes	198	140
No. of alarms	1155	790
No. of structural fires	245	155

6.3.1.1. Travel Time to Structural Fires. Option 1 reduces average travel time in its affected region by 0.90 minute, while Option 2 reduces average travel time in its affected region by 1.97 minutes. From a travel time point of view, the importance of this improvement is reflected in the number of structural fires that will receive this improved travel time. We can measure the total decrease in travel time to structural fires by taking the product of the change in travel time and the number of structural fires. Under Option 1, where the affected region had 245 structural fires, the total reduction in travel time is .90 minute x 245 structural fires = 220.5 minutes. Doing a similar calculation for Option 2, which had 155 structural fires in its affected region, the total reduction in travel time is 305.4 minutes. Thus, Option 2 has a significantly better effect on travel time to structural fires than does Option 1.

6.3.1.2. Coverage. From a coverage standpoint, the more alarm boxes that receive an improved travel time the better. One way to measure the magnitude of the change in coverage is to compute the product of the change in average travel time and the number of alarm boxes in the affected region. Under Option 1, coverage in the affected region improves by .90 minute x 198 boxes = 178.2 minutes. Under Option 2, coverage improves by 1.97 minutes x 140 boxes = 275.80 minutes. Again, Option 2 is significantly better than Option 1.

6.3.1.3. Hazards. The results presented above show that Option 2 is better in terms of both average travel time to structural fires and coverage. However, under Option 1 the truck company is located closer to the greater fire hazards in Demand Regions 4 and 2. The question of whether the difference in hazard level is great enough to offset the improvements in coverage and average travel time has no clear answer. The determination of which of the two options should be implemented must include a subjective evaluation of the relative hazards to which Truck 1 must respond from each location, consideration of the political constraints that enter into such a policy decision, as well as the quantitative measures calculated above.

6.3.2. Engines

A three-phased redeployment of Trenton's nine engine companies that results in a considerable improvement in travel times throughout the city was developed. The primary effect of Phase 1 is to reduce first- and second-due travel times in Regions 2 and 4. Phase 2 reduces average travel time in Region 1 but, more importantly, it reduces the number of boxes to which the first-due travel time is greater than three minutes and the second-due travel time is greater than four minutes. Phase 3 reduces the number of long first-due and second-due travel times in Region 6.

Figure 5 shows each of the moves on a map of Trenton. Table 5 summarizes the effect of the moves on travel times, showing the results for the final configuration as well as the results after each of the two intermediate steps. The final configuration results in a reduction in the average citywide first-due engine travel time from the current 1.51 minutes to 1.29 minutes and provides more balanced fire protection over the city. The greatest improvement is in Region 4, where the average travel time goes from 1.62 minutes to 1.24 minutes, a reduction of 23 percent. Regions 1 and 2 benefit almost as much. Average first-due travel time falls from 2.00 minutes to 1.65 minutes in Region 1 and from 1.41 minutes to 1.10 minutes in Region 2. In Region 6, average first-due travel time drops to 1.29 minutes from its current 1.53 minutes. The moves do not affect average first-due travel times in Regions 3 or 5, where the average travel times remain at 1.15 minutes and 1.19 minutes, respectively.

The three-phased redeployment results in a reduction in citywide maximum first-engine travel time of almost 30 percent, from 5.01 minutes to 3.57 minutes. The maximum first-engine travel times in Regions 1, 2, and 4 are reduced by approximately 1.5 minutes in each region, and the maximum time in Region 6 is cut by over one minute, from 4.17

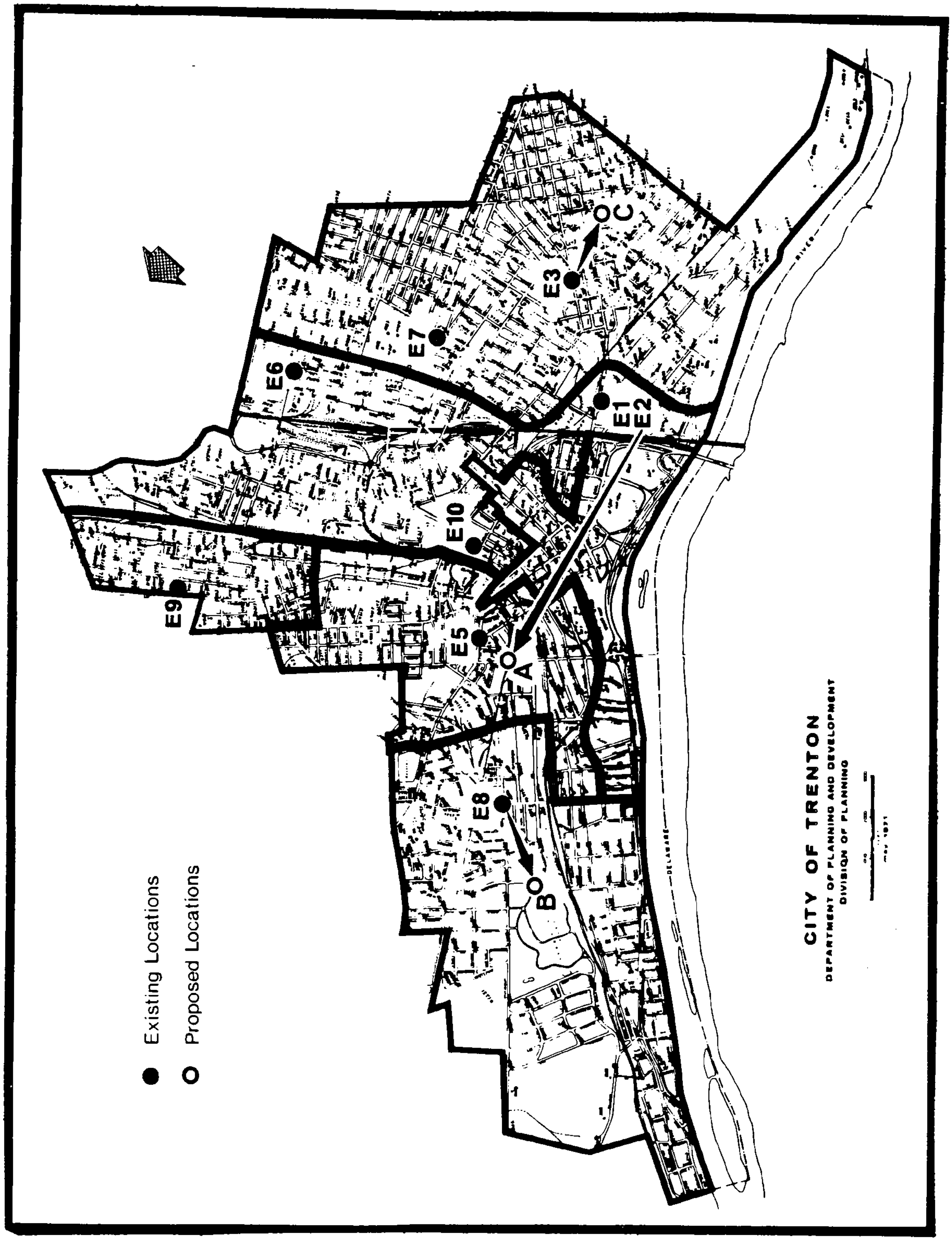


Figure 5. Proposed redeployment of engine companies.

Table 5

AVERAGE AND MAXIMUM ENGINE TRAVEL TIMES[a]

	Current Deployment		After Phase 1		After Phase 2		After Phase 3	
	Average Travel Time	Maximum Travel Time	Average Travel Time	Maximum Travel Time	Average Travel Time	Maximum Travel Time	Average Travel Time	Maximum Travel Time
First-Due								
Citywide	1.51	5.01	1.41	4.32	1.36	4.17	1.29	3.57
Region 1	1.41	3.35	1.09	1.91	1.10	1.91	1.10	1.91
Region 2	1.41	3.35	1.09	1.91	1.10	1.91	1.10	1.91
Region 3	1.15	1.85	1.15	1.85	1.15	1.85	1.15	1.85
Region 4	1.62	3.38	1.22	1.76	1.24	1.76	1.24	1.76
Region 5	1.19	1.80	1.19	1.80	1.19	1.80	1.19	1.80
Region 6	1.53	4.17	1.53	4.17	1.53	4.17	1.29	3.06
Second-Due								
Citywide	2.31	7.50	2.01	5.41	2.02	5.41	2.06	5.41
Region 1	3.82	7.50	2.97	5.41	3.01	5.41	3.01	5.41
Region 2	2.28	4.24	1.41	2.35	1.42	2.35	1.42	2.35
Region 3	2.79	4.03	2.79	4.03	2.79	4.03	2.79	4.03
Region 4	1.68	3.91	1.53	2.27	1.55	2.27	1.63	2.36
Region 5	1.82	2.69	1.73	2.68	1.73	2.68	1.80	2.68
Region 6	1.91	3.63	1.91	3.63	1.91	3.63	1.99	4.17

[a] All travel times in minutes.

minutes to 3.06 minutes. More importantly, only 8 of the 36 boxes that had first-due travel times of three minutes or more will still have such long travel times. Six of these boxes are located in Region 1 and two in Region 6.

Citywide travel time for second-due engines is also reduced. The average second-due travel time is improved from 2.31 minutes to 2.06 minutes. The greatest improvements occur in Regions 1 and 2. The average second-due travel time is reduced from 3.82 minutes to 3.01 minutes, or 21 percent, in Region 1 and from 2.28 minutes to 1.42 minutes, or 38 percent, in Region 2. Slight improvements also occur in Regions 4 and 5; there is no change in Region 3; and in Region 6 average second-due travel time is somewhat degraded, increasing from 1.91 minutes to 1.99 minutes (approximately 5 seconds, or 4 percent).

The citywide maximum second-due engine travel time is reduced by 28 percent, going from 7.50 minutes to 5.41 minutes. The greatest improvements occur in Regions 1, 2, and 4. In Region 6 the maximum travel time is increased by one-half minute, going from 3.63 minutes to 4.17 minutes. However, this higher maximum in Region 6 makes it comparable to two other regions of similar hazard, Regions 1 and 3, where the maximum second-engine travel times are 5.41 minutes and 4.03 minutes, respectively. The redeployment leaves only 23 alarm boxes in Trenton to which second-due engine travel time exceeds four minutes, compared to 53 alarm boxes prior to the redeployment. Of these 23 alarm boxes, 18 are located in Region 1, 2 in Region 3, and 3 in Region 6. These are the three regions with the lowest fire hazards and alarm densities in Trenton.

We complete the analysis of this three-phased redeployment of engine companies by looking at the region affected by the moves. For first-due engine response, the affected region consists of 295 boxes at which a total of 318 structural fires occurred in 1973. Average travel time in this region is reduced by 26 percent, going from 1.91 minutes to 1.41 minutes. Therefore, the total reduction in travel time to the affected boxes is 147.5 (.5 x 295) minutes, and the total reduction in travel time to structural fires is 159 (.5 x 318) minutes. Included in the 295 boxes in the affected region are 58 boxes to which average travel time is increased from 1.01 minutes to 1.38 minutes and 237 boxes to which travel time is reduced from an average of 2.13 minutes to an average of 1.42 minutes.

For second-due engine response, the affected region consists of 330 alarm boxes to which travel time is improved by an average of 18 percent, going from 2.65 minutes to 2.16 minutes. A total of 430 structural fires occurred in the second-due affected region. Therefore, the total reduction in travel time to boxes is 161.7 minutes, while the total reduction in travel time to structural fires is 210.7 minutes. Of the 330 boxes in the affected region, 117 receive degraded responses, the average travel time increasing from 1.55 minutes to 1.97 minutes. Average travel time to the other 213 boxes is improved by 30 percent, going from 3.25 minutes to 2.26 minutes. Table 6 summarizes the changes in both the first- and second-due affected regions.

Table 6

SUMMARY OF CHANGES IN THE AFFECTED REGIONS

Affected Region Measure	All Boxes	Degraded Boxes	Improved Boxes
First-Due			
(1) Reduction in average travel time (mins.)	0.50	-0.37	0.71
(2) No. of boxes	295	58	237
(3) No. of structural fires	318	86	232
Second-Due			
(4) Reduction in average travel time (mins.)	0.49	-0.42	0.99
(5) No. of boxes	330	117	213
(6) No. of structural fires	430	125	305

7. THE SITING MODEL DATA BASE

7.1. INTRODUCTION

The Firehouse Site Evaluation Model performs its calculations using detailed information about a set of points, which we have called alarm boxes, spread throughout the city. These may correspond to real alarm boxes or to grid coordinates, street intersections, tax blocks, etc. Each box, although represented as a point by a pair of grid coordinates, is associated with the alarms occurring in the area surrounding it. One box's area does not overlap with any other box's area and taken together, all of these areas should constitute the entire city. Each box should be associated with an area that is small enough so that the travel time to the box can be used as an estimate of the travel time to any point in the area. But the area should be large enough so that the number of boxes in manageable.* The data on boxes, along with data on fire companies and certain control and summary information, must be stored in a computer file accessible to the program. We call this file the siting model data base.

As currently written, the siting model program makes several assumptions about the way boxes and companies are identified. First, the program assumes that boxes are identified by integers in the range 1-9999. Leading zeros in box identifiers are ignored. Second, the program recognizes only two types of fire-fighting apparatus—engines (or pumpers) and ladders (or trucks). Engines are identified by an "E" followed by a number in the range 1-99. Ladders are identified by an "L" followed by a number in the range 1-99. Engine and ladder companies may share the same numeric identifier. For instance, a city might have an Engine Company No. 1 and a Ladder Company No. 1. To identify these companies to the program, the user refers to E1 and L1, respectively. Leading zeros in the numeric portion of company identifiers are ignored.

The siting model package includes a utility program, named DATGEN, that must be used to transform raw data on fire companies and boxes into a data base for the site evaluation program, named SITE. We will, therefore, describe the siting model data base in terms of the input requirements of the DATGEN program.

---

*The siting model has been used in cities having from 400 to 1000 boxes. The area covered by each box has ranged from .013 square mile to .021 square mile. The accuracy of the performance measures will decrease as the area per box increases, but the cost to run the model increases as the number of boxes is increased.

7.2. DATA REQUIREMENTS FOR THE DATGEN PROGRAM

The data required by DATGEN are described below in terms of the FORTRAN variables used to reference them. The data must be assembled into a computer-readable file. Table C.1 shows the format that must be used when the file is created. The computer storage characteristics of this file are described in Table D.1. The values of many control data items are constrained to be less than some limit by the dimensions of certain arrays in the program. These items are starred in the descriptions below. Their maximum values are given in Table D.3.

7.2.1. Control Information

(a) MAXCID* - The maximum numeric identifier of any company currently in service.

(b) MAXPID* - The maximum numeric identifier to be allowed for any company in a proposed configuration. Therefore, 2*MAXPID is the maximum number of companies (of all types) that can be considered in any proposed configuration. MAXPID must be greater than MAXCID. Identifier values between MAXCID and MAXPID are used for companies added to the current configuration.

(c) MAXDUE(1)* - The number of engine companies included in the alarm assignment lists (see below).

(d) MAXDUE(2)* - The number of ladder companies included in the alarm assignment lists. This value must be less than or equal to MAXDUE(1).

(e) NREG* - The number of demand regions into which the city has been partitioned.

(f) NBOX* - The number of boxes in the city for which detailed incidence, assignment, and geographic data are provided.

(g) RDSW - An indicator variable used to specify the method that the program will use to compute travel distance. A value of "0" or "1" means that travel distances between two locations will be estimated by the Euclidean (straight-line) distance between the points multiplied by a correction factor, RDFCT. This method will be used if no value is supplied for RDSW. A value of "2" or greater means that travel distances will be estimated by the sum of the absolute values of the differences between the x and y coordinates of the two points (right-angle distance). If this method of computing travel distance is chosen, specification of the angle THETA between the street grid and the coordinate axes will cause rotation of the coordinate axes coincident with the street grid. These two methods of estimating travel distances are discussed in Section 3.4 above.

(h) RDFCT - A factor to convert a Euclidean distance into an estimate of the travel distance. Based on experiments conducted in several cities, a value of 1.15 for this factor has been found to provide good estimates. If no value is provided for RDFCT then DATGEN will use 1.15.

(i) THETA - An angle (in degrees) through which coordinate axes are to be rotated to coincide with the street grid when the right-angle method of computing response distance is used. A positive value of THETA corresponds to counterclockwise rotation of coordinate axes. The default value of THETA is 0.

(j) SCALE - The number of grid coordinate units per mile. This value is needed in order to convert grid distances into miles; its default value is 1.0.

(k) Travel time function parameters (see Section 3.5):

1) A - the y-intercept of the linear part of the travel time function.
2) B - the slope of the linear part of the function.
3) C - the coefficient of the square-root part of the function.
4) XDIST - the cutoff distance (d) for the transition from the square-root to the linear part of the function.

If all of these values are omitted from the input data or are specified as having values of 0, default values of A = .65, B = 1.7, C = 2.1, and XDIST = .38 are used by DATGEN. (If a non-zero value is supplied for any response time parameter, then no defaults are supplied.)

(l) NCOMPS - The total number of companies in the city.

(m) KAACSW - An indicator variable used to determine whether alarm assignment lists are supplied in the data or whether they will be generated by DATGEN. A value of "0" or "1" indicates that the user is supplying the lists. A value of "2" or greater will cause DATGEN to generate the alarm assignment lists. The generated lists will place companies in increasing order of response distance, without regard to any barriers, such as railroad tracks, rivers, one-way streets, etc., that might increase the actual distance that a company would have to travel to a box.

7.2.2. Company Information

For each fire company in the city in the "current" configuration, the DATGEN program requires its name (Enn or Lnn) and its location on a grid of the city. The location is given by its x coordinate (CX) and y coordinate (CY).

7.2.3. Alarm Box Information

For each box identified in the city, the DATGEN program requires the following information:

(a) BOXNUM - The numeric identifier for the box. This can be an alarm box number, a tax block number, etc.

(b) X - The x coordinate of the box on a grid of the city.

(c) Y - The y coordinate of the box on a grid of the city.

(d) HZRGN - The number of the demand region to which the box belongs (see Section 2.2.1).

(e) TRGT - A binary flag to indicate whether the box represents a target hazard or not. A value of "1" indicates the presence and a "0" indicates the absence of a target hazard (see Section 2.2.4).

(f) BXTLRM - The total number of alarms that occurred in the area represented by the box during the previous year (or during any time period for which the data are available).

(g) BXSLRM - The total number of structural fires that occurred in the area represented by the box during the same time period used for the total number of alarms.

(h) ID (*,1)[+] - The alarm assignment list containing the identification numbers of the engine companies designated to respond to an alarm occurring at the box, in increasing order of travel time (or distance). If the user requests that DATGEN generate alarm assignment lists (see KAACSW above), then no values for ID need be supplied.

[+] The asterisk notation in an array reference refers to the entire range of the corresponding subscript; in this case, the reference is to all rows of column 1 of array ID.

(i) ID (*,2) - The alarm assignment list containing the identification numbers of the <u>ladder</u> companies designated to respond to an alarm occurring at the box, in increasing order of travel time (or distance). If the user requests that DATGEN generate alarm assignment lists, then no values for ID need be supplied.

8. SETTING UP THE SITING MODEL

8.1. GENERAL PROCEDURES

The procedure for preparing the Firehouse Site Evaluation Model for use on a computer system consists of three phases. These are: data collection and assembly, program installation, and data base generation.

In the first phase, values for the data items described in Section 7 must be collected and assembled into a computer-readable file.

The program installation phase involves setting up the two programs in the siting model package (named DATGEN and SITE) on the user's computer system. This includes setting parameters in COMMON blocks to describe the file conventions of the user's computer system, altering array dimensions if this is required, and compiling the programs. Complete instructions for changing parameters and dimensions in both programs are given below in this section.

Both DATGEN and SITE are written in ANSI FORTRAN with two extensions: (1) generalized subscripts; and (2) literals enclosed in apostrophies in FORMAT statements. Therefore, the siting model can be run at any installation having a FORTRAN compiler with these characteristics.

In phase 3 of the setting-up process, the DATGEN program is used to transform the raw data, which the user has collected and placed in a computer-readable file, into a data base for SITE, which is the siting model program. Section 8.1.1 describes how to run the DATGEN program.

After DATGEN is run, external input and output files (including the data base file generated by DATGEN) must be associated with FORTRAN file reference numbers for the SITE program. Once this is done, the siting model is ready to be used. Details on how to use the SITE program are given in Section 5.

8.1.1. Using the DATGEN Program

The DATGEN program transforms grid coordinate, alarm assignment, and alarm incident data into a data base that can be used by the SITE program. The file containing the input data is named DATIN (this is the name of the FORTRAN variable that contains the FORTRAN file reference number for the file); its device requirements are described in Table D.1 and its format is given in Table C.1. The file containing the generated data base is called DATOUT; its device requirements are described in Table D.1 and its format in Table C.2.

Before executing the DATGEN program, its external files must be associated with the FORTRAN file reference numbers used by the program (see Section 8.3 and Table D.1).

DATGEN can be run in either batch or interactive mode. Figure 6 shows output from a sample run in interactive mode. The program begins execution by printing control information (see Section 7.2.1). Most of this information must be supplied by the user in DATIN; the program will supply some control information by default. After printing the control information, the program reads an indicator that the user must supply, which tells it whether to continue execution and generate the data base or to stop execution because the user has detected some problem with the control information and wishes to correct it before re-executing the program. When using the program in interactive mode, the user must respond 'Y' or 'YES' to the message 'CONTINUE?' typed by the program to cause the program to generate the data base; any other response not beginning with the letter 'Y' will cause the program to terminate execution.

In batch mode the DATGEN program should be run at least twice. In the first run, the SYSIN (standard card input) file should consist of a single card that contains a character other than a 'Y' in column 1. This will cause the program to stop after printing the control information. Then, by examining the output, the user can determine whether or not the control information is correct. After verifying the correctness of the control information (additional runs might be required to reach this stage if errors are detected), a card that contains a 'Y' in column 1 can be substituted and the program run again to create the data base.

DATGEN performs several kinds of consistency checks on the data supplied to it, so that it can generate a data base that requires no error checking by the SITE program. Some errors are considered fatal and DATGEN will not complete data base generation if they are detected; others are corrected by DATGEN. Self-explanatory messages are printed for all error conditions. The fatal error conditions are:

- MAXDUE(2) greater than MAXDUE(1)
- Box record out of numerical order
- Invalid demand region in a box record
- Invalid company identification number in a company coordinate record.

```
                    NEW YORK CITY - RAND INSTITUTE
                    FIRE HOUSE SITE EVALUATION MODEL

                         DATA BASE GENERATION

DATA BASE PARAMETERS

NUMBER OF BOXES:   74
NUMBER OF DEMAND REGIONS:  6
MAXIMUM # ENGINES ON ALARM ASSIGNMENT CARD:  9
MAXIMUM # LADDERS ON ALARM ASSIGNMENT CARD:  4
MAXIMUM ID OF EXISTING COMPANIES: 10
MAXIMUM ID OF COMPANIES TO BE ADDED: 20
NUMBER OF COMPANIES IN CITY:  13
RESPONSE TIME SPLINE FUNCTION PARAMETERS: A = 0.533, B = 1.719, C = 1.914
                                          CUTOFF = 0.310
SCALE: 52.8 COORDINATE UNITS/MILE
RESPONSE DISTANCE DETERMINED FROM EUCLIDIAN DISTANCE WITH FACTOR 1.15
ALARM ASSIGNMENT LISTS ARE USER-SUPPLIED.

CONTINUE? Y


*** INVALID LADDER COMPANY FOR BOX 2151
    CURRENT ASSIGNMENT:     1  2  3  0
    CORRECTED ASSIGNMENT:   1  4  2  3

*** INVALID ENGINE COMPANY FOR BOX 2219
    CURRENT ASSIGNMENT:     1  5 10  8  0  0  0  0  0
    CORRECTED ASSIGNMENT:   5  8 10  1  2  3  7  9  6

*** INVALID LADDER COMPANY FOR BOX 2219
    CURRENT ASSIGNMENT:     1  4  0  0
    CORRECTED ASSIGNMENT:   1  4  3  2

*** INVALID ENGINE COMPANY FOR BOX 8232
    CURRENT ASSIGNMENT:     2  1 10  3  5  7  6  0  0
    CORRECTED ASSIGNMENT:   1  2  3 10  5  7  8  6  9

*** INVALID ENGINE COMPANY FOR BOX 8239
    CURRENT ASSIGNMENT:    10  2  1  9  7  5  0  0  0
    CORRECTED ASSIGNMENT:   1  2  3 10  7  5  6  8  9

*** INVALID LADDER COMPANY FOR BOX 8239
    CURRENT ASSIGNMENT:     4  1  0  0
    CORRECTED ASSIGNMENT:   1  4  2  3

DATA BASE GENERATION SUCCESSFUL
```

Figure 6. Sample DATGEN run in interactive mode.

A non-fatal error occurs if the identification number of a non-existent company appears in an alarm assignment list for a box. In this case, DATGEN ignores the list and generates a completely new alarm assignment list with companies in increasing order of distance from the box. An error message showing the user-supplied and program-corrected alarm assignment lists is printed and execution continues (see Fig. 6).

When a data base has been successfully generated, the message "DATA BASE GENERATION SUCCESSFUL" is printed and execution terminates.

8.2. THE STRUCTURE AND LOGICAL FLOW OF THE TWO PROGRAMS

8.2.1. DATGEN

DATGEN consists of a main program and four subprograms, one of which is a BLOCK DATA subprogram. These five modules must be compiled and combined to form a complete program unit. The specifics of this process will vary according to the particular computer system on which the program is to be implemented and for this reason are not included in this document.

DATGEN first reads the control information, such as the number of boxes in the city, from the input file (DATIN) and writes this information and a title on the print file (SYSOUT). Company coordinates are then read from DATIN and saved for use in computing response distances. Then all box data are read. Box numbers and coordinates are saved in arrays and the remaining box information is written on a scratch file (SCRTCH). Box and company coordinates are converted to miles and coordinate axes rotated as specified by the value of THETA. Control information, box numbers, and box coordinates are written on DATOUT.

In the main loop of DATGEN (the DO 199 loop in the DATGEN program listing), a record is read from SCRTCH for each box in the data base. Each record contains a demand region identification, target hazard indicator, total and structural alarm incident counts, and a list containing alarm assignment information. A travel distance and a travel time to the box are computed for each company on the list; these are accumulated for the appropriate summary statistics. The travel time and distance weighted by both total and structural alarms are also accumulated. A record is written on DATOUT for each box, containing demand region, target hazard, incident, alarm assignment, and travel distance information. If an alarm assignment list contains an invalid company identifier, the list is replaced with a program-generated list containing the closest companies.

Finally, summary statistics are computed and written on DATOUT.

8.2.2. SITE

SITE consists of a main program and thirteen subprograms, one of which is a BLOCK DATA subprogram. These fourteen modules must be compiled and combined to form a complete program unit. Appendix G provides a cross-referenced listing of all program segments, including subprograms and common blocks.

The SITE program operates in a continuous loop, which is terminated only when a user Exit command ("E") is encountered. The loop starts by reading the control information and lists of box numbers and their coordinates from the input file (IUNIT).

Subroutine INPUT is then called to read and encode the user's commands (commands are saved in a coded form). Control is returned to the main program by any command that requires immediate action, such as Output ("O") or Cancel changes ("C"). After control is returned to the main program, it branches to a routine to carry out the last command entered. The bulk of the program's code carries out the Output command.

Whenever the user requests output, SITE reads through the data base and applies the configuration changes that the user has specified to each alarm box record in the data base. Then, the program calculates the statistics that were requested in the Output command. Mapping, alarm assignment, and new data base information are written if requested.

After all the records in the data base have been processed, the statistics for the current configuration are read from the data base and the output requested by the user is printed. Upon completion of printing, the program returns to the beginning of the main loop, where the data base file is rewound, the box number, coordinate, and control information are reread, and INPUT is called to accept the next command sequence.

8.3. COMPUTER SYSTEM-DEPENDENT PARAMETERS

The DATGEN and SITE programs reference all input and output files through FORTRAN variables rather than explicit unit numbers to allow assignment of unit numbers that conform to the conventions of a particular computer system. The file reference variables, except for IUNIT and OUNIT in SITE, appear in COMMON/SYSTEM/ in both DATGEN and SITE (note that the contents of SYSTEM are not identical in both programs) and may be changed from their default values by altering the appropriate DATA statements in the BLOCK DATA subprograms.

Tables D.1 and D.2 describe the input and output data files used by DATGEN and SITE, respectively. File names are the FORTRAN variables used by the programs to reference the files. The unit numbers shown are those in the distributed program.

8.4. CITY SIZE- AND DATA-DEPENDENT DIMENSIONS

8.4.1. Arrays Common to DATGEN and SITE

Most of the information in the data base created by the DATGEN program is in array form. The siting model programs have been written so that the dimensions of their arrays can be larger than the dimensions required for a particular city. Table D.4 provides minimum dimensions for the arrays created for SITE by DATGEN in terms of the "control variables" that are used to reference the arrays.

Table D.3 provides definitions of the control variables in terms of parameters of the city. It will be necessary to alter DIMENSION statements in DATGEN and SITE only if one or more attributes of the city exceed the default maximum of the corresponding control variable in Table D.3. Further, it is necessary to alter only those array dimensions that depend on the control variables whose maximum default value is exceeded. For instance, if the number of boxes in the city is greater than 750, then the dimensions of arrays in DATGEN and SITE whose minimum dimensions depend on NBOX will have to be increased at least to the number of boxes in the city.

Table D.4 indicates that some arrays are in common blocks. Refer to Appendix G to find the subprograms in which common blocks occur. Arrays must be identically dimensioned in all occurrences of the common blocks that contain them. Note that many arrays are entered into common blocks by EQUIVALENCE statements (see Section 8.4.4 below).

8.4.2. Arrays in DATGEN Only

There are two arrays that DATGEN uses for computing response distances that are not contained in the siting model data base. These arrays are CX(MAXCID,2) and CY(MAXCID,2), which contain the x and y coordinates of the companies in the current configuration. These are real variables and appear only in MAIN.

8.4.3. Arrays in SITE Only

Table D.5 describes the city size-dependent arrays that appear only in the SITE program in terms of the control variables defined in Table D.3. The comments of Section 8.4.1 regarding array dimensions apply to these arrays as well.

8.4.4. EQUIVALENCE Statements

For each type of statistic in the output reports, one array is used to hold information for both company types and for all response levels. Since there are generally about twice as many engine companies as ladder companies in a city, approximately 25 percent of the storage for statistics would be wasted if storage were reserved for an equal number of engine and ladder company response levels. For instance, if alarm assignment data were given for ten engines and five ladders, all storage allocated for sixth- through tenth-due ladder statistics would go unused. To minimize storage requirements for these statistics, EQUIVALENCE statements are used in various common blocks and in the MAIN program of SITE to overlay unused parts of arrays. Letting I be one greater than the number of ladders in the alarm assignment data, the EQUIVALENCE statements (see the first and second pages of the SITE program listing in Appendix E) should have the following form:

For COMMON/NEWREG/:

```
      EQUIVALENCE (NRRD,RHIST(1,1,I,2)),(NRWRD,NRRD(1,I,2)),
     C (NRRT,NRWRD(1,I,2)),(NRWRT,NRRT(1,I,2)),
     C (NRMRT,NRWRT(1,I,2)),(NRSWRD,NRMRT(1,I,2)),
     C (NRSWRT,NRSWRD(1,I,2)).
```

For COMMON/OLDREG/:

```
      EQUIVALENCE (AFTLRM,AFNBOX(1,I,2)),(AFSLRM,AFTLRM(1,I,2)),
     C (ABXPTR,AFSLRM(1,I,2)),(ORRD,ABXPTR(1,I,2)),
     C (ORWRD,ORRD(1,I,2)),(ORRT,ORWRD(1,I,2)),
     C (ORWRT,ORRT(1,I,2)),(ORMRT,ORWRT(1,I,2)),
     C (ORSWRD,ORMRT(1,I,2)),(ORSWRT,ORSWRD(1,I,2)).
```

For COMMON/COMP/:

```
      EQUIVALENCE (NCMRT,NCRT(1,I,2)),(NCTLRM,NCMRT(1,I,2)),
     C (NCSLRM,NCTLRM(1,I,2)),(NCBOX,NCSLRM(1,I,2)),
     C (CBXPTR,NCBOX(1,I,2)),(OCRT,CBXPTR(1,I,2)),
     C (OCMRT,OCRT(1,I,2)),(OCTLRM,OCMRT(1,I,2)),
     C (OCSLRM,OCTLRM(1,I,2)),(OCBOX,OCSLRM(1,I,2)).
```

For MAIN:

```
      EQUIVALENCE (OTGRT,OTGRD(1,I,2)),(NTGRD,OTGRT(1,I,2)),
     C (NTGRT,NTGRD(1,I,2))
      EQUIVALENCE (LNBUF,NTGRT(1,I,2)), (MXD,MAXDUE(1)),
     C (MXD2,MAXDUE(2))
```

As distributed, the EQUIVALENCE statements are coded with I replaced by "6." If the number of ladders in the alarm assignment data is less than or equal to five, there is no need to change the EQUIVALENCE statements. This "recipe" will work for all cases in which MAXDUE(1) is greater than MAXDUE(2) (see Section 7.2.1). If MAXDUE(1) and MAXDUE(2) are equal, the EQUIVALENCE statements should be removed and the equivalenced variables placed directly in common blocks. DATGEN will not generate a data base if MAXDUE(1) is less than MAXDUE(2).

8.5. STORAGE-LIMITED DIMENSIONS AND PARAMETERS

There are two arrays in the SITE program whose storage is used dynamically at run time. If core storage is readily available, the dimensions of these arrays may be increased to provide greater flexibility and a possible reduction in cost at run time. If storage is tight, their dimensions may be reduced with the penalty of decreased run-time flexibility and possibly increased dollar cost.*

The dimensions of the arrays in Table D.6 should be equal to the values of the parameters used to describe them. The parameter NLISTS occurs in COMMON/LISTS/ and BETAMX occurs in COMMON/CV/. Their values may be altered by changing the appropriate DATA statement in the BLOCK DATA subprogram. NLISTS is the number of separate lists of alarm boxes that SITE can maintain simultaneously. Its default value is 10. The minimum value of NLISTS is 2, because the program uses this storage for temporary storage of box coordinates. Lists are used only during the execution of an "O" command. The number of lists that the program uses at any one time is given by:

LSW*IDUE*NCTYPE *(CSW+ASW) + 2*MSW,

where

- LSW is 1 or 0, depending upon whether or not lists of box numbers have been requested;
- IDUE is the highest response level for which output has been requested;
- NCTYPE is the number of company types for which output has been requested;
- CSW is 1 or 0, depending upon whether or not company response area output has been requested;
- ASW is 1 or 0, depending upon whether or not affected region output has been requested;
- MSW is 1 or 0, depending upon whether or not mapping output has been requested.

BETAMX is the number of words of storage allocated to saving changes to the current configuration entered by the user. Its default value is 200.

---

*Increased dollar cost can result from having to issue multiple "O" commands to get requested box lists if insufficient storage is available to accumulate all lists in one pass over the data base.

8.6. SETTING UP AND USING THE SITE PROGRAM IN BATCH MODE

Although the SITE program, as distributed, can be run in batch mode, three changes in the program will make the output easier to understand and better to look at. All three changes are in SUBROUTINE TOKEN (see Appendix E).

To modify the SITE program for batch mode operation:

(1) remove the statement WRITE(SYSOUT,2),

(2) after the statement READ(SYSIN,1)(CARD(I),I=1,80) add the statement WRITE(SYSOUT,3)(CARD(I),I=1,80),

(3) add the statement 3 FORMAT(1H ,80A1).

To run the program in batch mode, simply assign the SYSIN file to a card reader or equivalent device, assign the SYSOUT file to a line printer, punch the user commands on cards (free-form), and place them in the SYSIN file.

Appendix A

GLOSSARY

Affected Region	The set of alarm boxes for which the travel time of the first-due unit has changed as a result of a change in deployment. Affected regions can also be defined for second-due units, etc.
Alarm Assignment List	A list associated with every alarm box that specifies the companies of each type that are closest to the box, in increasing order of response distance.
Alarm Boxes	The generic name used to refer to the discrete set of points spread throughout a city that represent locations where alarms are assumed to occur. The set of points may correspond, for example, to actual street alarm boxes, "phantom" boxes, grid coordinates, street intersections, or the centroids of tax blocks. Each alarm box, although represented as a point in the model, is associated with the alarm incidence in the area surrounding it. Since travel times are calculated only to alarm boxes, the density of these selected locations should be high enough so that the travel time to an alarm box will provide a good estimate of the travel time to the surrounding area. The set of all of these areas should be the entire city.
Company	Used interchangeably with "fire company" to specify any fire department vehicle and its complement of firemen. See also "Engine Company" and "Ladder Company."
Company Response Area	See "First-Due Response Area."
Demand Region	An area of a city that contains a homogeneity of fire hazards to life and property, potential fire-fighting problems, and alarm incidence. Population density, land use, housing stock, natural barriers, and alarm incidence should all be taken into consideration in determining the alarm boxes to place in each demand region.

Deployment (of Fire Companies) — The manner in which fire companies are located and dispatched. Deployment issues can be divided into two groups: strategic and tactical. Tactical issues are concerned with the dispatching of fire-fighting units to alarms and the relocation of available units into empty firehouses to fill temporary gaps that develop in coverage. Strategic issues are concerned with the number of companies to have on duty, how they should be manned, and where they should be located. The Firehouse Site Evaluation Model is most helpful in analyzing strategic deployment policies.

Engine Company — A group of firemen who man a pumper and have the responsibility of connecting a hose to a fire hydrant and delivering water onto a fire.

First-Due Company — The engine (ladder) company listed first on the alarm assignment list for a given alarm box. The first-due company is generally the nearest in distance or travel time to the box. The second-due company is the second listed, etc.

First-Due Response Area — The set of alarm boxes to which a given company is closest. Similar definitions apply for a company's second-due response area, etc.

Ladder Company — A group of firemen who man a truck equipped with an aerial ladder. The truck also carries a large assortment of emergency tools, and the men are responsible for forcible entry, rescue, ventilation, and salvage work.

Target Hazard	An alarm box identified by the user as more important than others for receiving a rapid response. The siting model output provides information on travel times to each target hazard and, overall, to the set of target hazards.
Travel Time	The time interval between departure of a fire company from its firehouse and its arrival at the closest alarm box to the incident.

Appendix B

AN ANNOTATED SAMPLE TERMINAL SESSION

```
                    NEW YORK CITY - RAND INSTITUTE
                   FIRE HOUSE SITE EVALUATION MODEL

COMMAND? M E10-1416,E6-0908[a]

COMMAND? D E2[b]

COMMAND? A L-1416[c]
```

[a]The "M" command moves Engines 10 and 6 to the locations of boxes 1416 and 908, respectively. As with all Move, Add, and Delete commands, the program takes no action when the command is entered, but stores the changes for use when an "O" (Output) command is executed.

[b]The "D" command removes Engine 2 from service.

[c]The "A" command adds a ladder company at the location of box 1416.

```
COMMAND? P [d]

 1)   MOVE ENGINE 10 TO BOX 1416
      MOVE ENGINE  6 TO BOX  908

 2)   DELETE ENGINE  2               [e]

 3)   ADD LADDER 11 AT BOX 1416

COMMAND? O D=2 [f]
```

[d] The "P" command causes all of the change commands stored so far to be printed out.

[e] Output from the "P" command shows the changes that will be processed when the next Output command is executed. These changes define the proposed configuration in terms of the current configuration. The change commands are numbered in the order they were entered. The changes within each command are printed one to a line. Note that the ladder added at box 1416 has been named Ladder 11.

[f] The "O" command requests output comparing the current and proposed configurations. The default values of all parameters except D are used (see Section 5.2.6). Output will, therefore, be produced for first- and second-due engines and ladders for citywide, demand, affected, hazard, and company response regions. Travel times and distances will be weighted by structural alarms, mapping output will not be produced, and box lists and new alarm assignment lists will not be produced. A new data base will not be written.

```
1ST DUE ENGINE RESPONSE [g]

CITYWIDE   ( 398 BOXES)  CURRENT   PROPOSED ( 4730 ALARMS, 1872 STRUCTS.)   [h]
AV. TR.T.                  1.98      1.96
AV. TR.D.                  0.74      0.73
AV. TR.T. TO STRUCTURALS   1.45      1.46
AV. TR.D. TO STRUCTURALS   0.42      0.43
MAX TR.T.                  3.80      3.80

REGION  1  (  71 BOXES)  CURRENT   PROPOSED ( 2215 ALARMS,  927 STRUCTS.)   [i]
AV. TR.T.                  1.51      1.41
AV. TR.D.                  0.46      0.40
AV. TR.T. TO STRUCTURALS   1.35      1.26
AV. TR.D. TO STRUCTURALS   0.36      0.31
MAX TR.T.                  2.52      2.52

REGION  2  (  58 BOXES)  CURRENT   PROPOSED (  254 ALARMS,   86 STRUCTS.)
AV. TR.T.                  2.09      2.09
AV. TR.D.                  0.80      0.80
AV. TR.T. TO STRUCTURALS   1.69      1.69
AV. TR.D. TO STRUCTURALS   0.57      0.57
MAX TR.T.                  3.59      3.59

REGION  3  ( 173 BOXES)  CURRENT   PROPOSED ( 1211 ALARMS,  492 STRUCTS.)
AV. TR.T.                  2.29      2.14
AV. TR.D.                  0.93      0.83
AV. TR.T. TO STRUCTURALS   1.60      1.65
AV. TR.D. TO STRUCTURALS   0.51      0.54
MAX TR.T.                  3.80      3.80

REGION  4  (  96 BOXES)  CURRENT   PROPOSED ( 1050 ALARMS,  367 STRUCTS.)
AV. TR.T.                  1.70      1.98
AV. TR.D.                  0.57      0.74
AV. TR.T. TO STRUCTURALS   1.45      1.67
AV. TR.D. TO STRUCTURALS   0.43      0.56
MAX TR.T.                  2.47      3.43
```

[g] The heading indicates that first-due engine statistics follow.

[h] Citywide first-due engine statistics for the current and proposed configurations.

[i] Demand Region 1 first-due engine statistics for the current and proposed configurations. Statistics for the other three demand regions follow.

```
DISTRIBUTION OF TRAVEL TIMES TO BOXES - CITYWIDE

TR TIME  0.5  1.0  1.5  2.0  2.5  3.0  3.5  4.0
# BOXES     18   97  110   85   46   35    7
```
[j]

```
DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  1

TR TIME  0.5  1.0  1.5  2.0  2.5  3.0
# BOXES      8   37   20    5    1
```
[k]

```
DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  2

TR TIME  0.5  1.0  1.5  2.0  2.5  3.0  3.5  4.0
# BOXES      2   11   17   11    8    8    1
```

```
DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  3

TR TIME  0.5  1.0  1.5  2.0  2.5  3.0  3.5  4.0
# BOXES      5   27   46   44   26   19    6
```

```
DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  4

TR TIME  0.5  1.0  1.5  2.0  2.5  3.0  3.5
# BOXES      3   22   27   25   11    8
```

[j] The histogram of first-due engine travel times to all boxes in the city under the proposed configuration.

[k] The histogram of first-due engine travel times to boxes in Region 1 under the proposed configuration. Travel time histograms for the other three demand regions in the city follow.

2ND DUE ENGINE RESPONSE[1]

CITYWIDE (398 BOXES)	CURRENT	PROPOSED (4730 ALARMS, 1872 STRUCTS.)
AV. TR.T.	2.84	2.75
AV. TR.D.	1.25	1.20
AV. TR.T. TO STRUCTURALS	2.03	2.11
AV. TR.D. TO STRUCTURALS	0.77	0.82
MAX TR.T.	5.27	5.06

REGION 1 (71 BOXES)	CURRENT	PROPOSED (2215 ALARMS, 927 STRUCTS.)
AV. TR.T.	1.93	1.94
AV. TR.D.	0.71	0.72
AV. TR.T. TO STRUCTURALS	1.69	1.78
AV. TR.D. TO STRUCTURALS	0.57	0.62
MAX TR.T.	3.37	3.34

REGION 2 (58 BOXES)	CURRENT	PROPOSED (254 ALARMS, 86 STRUCTS.)
AV. TR.T.	2.99	2.96
AV. TR.D.	1.34	1.32
AV. TR.T. TO STRUCTURALS	2.43	2.43
AV. TR.D. TO STRUCTURALS	1.01	1.01
MAX TR.T.	5.06	5.06

REGION 3 (173 BOXES)	CURRENT	PROPOSED (1211 ALARMS, 492 STRUCTS.)
AV. TR.T.	3.26	3.09
AV. TR.D.	1.50	1.40
AV. TR.T. TO STRUCTURALS	2.39	2.62
AV. TR.D. TO STRUCTURALS	0.98	1.12
MAX TR.T.	5.27	5.02

REGION 4 (96 BOXES)	CURRENT	PROPOSED (1050 ALARMS, 367 STRUCTS.)
AV. TR.T.	2.68	2.62
AV. TR.D.	1.15	1.12
AV. TR.T. TO STRUCTURALS	2.31	2.16
AV. TR.D. TO STRUCTURALS	0.94	0.85
MAX TR.T.	3.84	3.76

[1] Second-due engine statistics.

DISTRIBUTION OF TRAVEL TIMES TO BOXES - CITYWIDE

TR TIME	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5
# BOXES		11	71	93	79	71	32	23	16	2

DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION 1

TR TIME	1.0	1.5	2.0	2.5	3.0	3.5
# BOXES		7	38	18	5	3

DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION 2

TR TIME	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5
# BOXES		3	7	17	8	5	4	5	8	1

DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION 3

TR TIME	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5
# BOXES		1	10	35	36	41	23	18	8	1

DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION 4

TR TIME	1.5	2.0	2.5	3.0	3.5	4.0
# BOXES		16	23	30	22	5

1ST DUE ENGINE RESPONSE [m]

	CO.	BOXES		AV. TR.T.		MAX TR.T.		ALARMS		STRUCTURALS	
		CURR.	PROP.	CURR.	PROP.	CURR.	PROP.	CURR.	PROP.	CURR.	PROP.
*	1	14	22	1.71	1.48	2.49	2.25	492	692	199	268
*	2	9	0	1.17	0.00	1.51	0.00	356	0	141	0
*	3	98	96	2.40	2.38	3.80	3.80	325	373	124	150
*	5	45	48	1.70	1.69	2.40	2.40	349	313	116	106
*	6	56	41	1.72	1.95	2.47	3.43	776	842	271	327
*	7	17	18	1.27	1.29	1.78	1.78	642	693	286	314
*	8	26	48	1.60	1.95	2.52	3.10	650	879	281	342
	9	58	58	2.09	2.09	3.59	3.59	254	254	86	86
*	10	75	67	2.15	1.81	3.55	2.73	886	684	368	279

[m] First-due response area statistics for engine companies. Asterisks denote companies whose response areas are different in the current and proposed configurations.

2ND DUE ENGINE RESPONSE

	CO.	BOXES		AV. TR.T.		MAX TR.T.		ALARMS		STRUCTURALS	
		CURR.	PROP.	CURR.	PROP.	CURR.	PROP.	CURR.	PROP.	CURR.	PROP.
*	1	91	48	2.94	2.38	4.91	3.68	1650	1016	700	430
*	2	23	0	1.81	0.00	3.06	0.00	621	0	256	0
*	3	16	39	2.36	3.48	3.09	5.02	254	422	93	185
*	5	7	43	1.77	2.78	2.10	4.76	43	476	14	173
*	6	55	91	2.85	2.60	3.84	3.76	560	1175	204	463
*	7	73	94	2.87	2.71	5.06	5.06	389	888	147	313
*	8	54	11	2.38	1.77	3.70	2.09	948	464	356	195
*	9	5	56	1.67	3.04	1.86	4.58	71	78	29	30
*	10	74	16	3.62	2.85	5.27	4.65	194	211	73	83

[n] Second-due response area statistics for engine companies.

```
1ST DUE ENGINE RESPONSE

CITYWIDE

  64 DEGRADED BOXES       CURRENT  PROPOSED ( 1091 ALARMS,  414 STRUCTS.)
AV. TR.T.                   1.57     2.23
AV. TR.D.                   0.50     0.89
AV. TR.T. TO STRUCTURALS    1.32     1.92
AV. TR.D. TO STRUCTURALS    0.35     0.70
MAX TR.T.                   2.39     3.43

  86 IMPROVED BOXES       CURRENT  PROPOSED ( 1093 ALARMS,  441 STRUCTS.)
AV. TR.T.                   2.27     1.69
AV. TR.D.                   0.91     0.57
AV. TR.T. TO STRUCTURALS    1.88     1.37
AV. TR.D. TO STRUCTURALS    0.68     0.38
MAX TR.T.                   3.55     2.73

 150 AFFECTED BOXES       CURRENT  PROPOSED ( 2184 ALARMS,  855 STRUCTS.)
AV. TR.T.                   1.97     1.92
AV. TR.D.                   0.74     0.71
AV. TR.T. TO STRUCTURALS    1.61     1.63
AV. TR.D. TO STRUCTURALS    0.52     0.53
MAX TR.T.                   3.55     3.43
```

[o] First-due engine citywide affected region statistics. Affected region statistics by demand region follow.

REGION 1

2 DEGRADED BOXES	CURRENT	PROPOSED (75 ALARMS, 35 STRUCTS.)
AV. TR.T.	1.33	1.61
AV. TR.D.	0.35	0.52
AV. TR.T. TO STRUCTURALS	1.34	1.66
AV. TR.D. TO STRUCTURALS	0.36	0.55
MAX TR.T.	1.35	1.70

12 IMPROVED BOXES	CURRENT	PROPOSED (386 ALARMS, 167 STRUCTS.)
AV. TR.T.	1.93	1.30
AV. TR.D.	0.71	0.34
AV. TR.T. TO STRUCTURALS	1.82	1.27
AV. TR.D. TO STRUCTURALS	0.65	0.32
MAX TR.T.	2.49	1.78

14 AFFECTED BOXES	CURRENT	PROPOSED (461 ALARMS, 202 STRUCTS.)
AV. TR.T.	1.84	1.35
AV. TR.D.	0.66	0.36
AV. TR.T. TO STRUCTURALS	1.74	1.34
AV. TR.D. TO STRUCTURALS	0.60	0.36
MAX TR.T.	2.49	1.78

REGION 3

21 DEGRADED BOXES	CURRENT	PROPOSED (507 ALARMS, 228 STRUCTS.)
AV. TR.T.	1.39	1.85
AV. TR.D.	0.39	0.66
AV. TR.T. TO STRUCTURALS	1.29	1.79
AV. TR.D. TO STRUCTURALS	0.33	0.63
MAX TR.T.	2.10	2.25

58 IMPROVED BOXES	CURRENT	PROPOSED (413 ALARMS, 145 STRUCTS.)
AV. TR.T.	2.47	1.83
AV. TR.D.	1.03	0.65
AV. TR.T. TO STRUCTURALS	2.13	1.52
AV. TR.D. TO STRUCTURALS	0.83	0.46
MAX TR.T.	3.55	2.73

79 AFFECTED BOXES	CURRENT	PROPOSED (920 ALARMS, 373 STRUCTS.)
AV. TR.T.	2.18	1.84
AV. TR.D.	0.86	0.66
AV. TR.T. TO STRUCTURALS	1.61	1.68
AV. TR.D. TO STRUCTURALS	0.52	0.56
MAX TR.T.	3.55	2.73

REGION 4

41 DEGRADED BOXES	CURRENT	PROPOSED (509 ALARMS, 151 STRUCTS.)
AV. TR.T.	1.68	2.45
AV. TR.D.	0.56	1.02
AV. TR.T. TO STRUCTURALS	1.36	2.18
AV. TR.D. TO STRUCTURALS	0.37	0.86
MAX TR.T.	2.39	3.43

16 IMPROVED BOXES	CURRENT	PROPOSED (294 ALARMS, 129 STRUCTS.)
AV. TR.T.	1.83	1.49
AV. TR.D.	0.65	0.45
AV. TR.T. TO STRUCTURALS	1.66	1.33
AV. TR.D. TO STRUCTURALS	0.55	0.35
MAX TR.T.	2.47	2.21

57 AFFECTED BOXES	CURRENT	PROPOSED (803 ALARMS, 280 STRUCTS.)
AV. TR.T.	1.72	2.18
AV. TR.D.	0.59	0.86
AV. TR.T. TO STRUCTURALS	1.50	1.79
AV. TR.D. TO STRUCTURALS	0.45	0.62
MAX TR.T.	2.47	3.43

```
2ND DUE ENGINE RESPONSE

CITYWIDE

 151 DEGRADED BOXES       CURRENT  PROPOSED ( 2063 ALARMS,  821 STRUCTS.)
AV. TR.T.                   2.61     3.00
AV. TR.D.                   1.11     1.35
AV. TR.T. TO STRUCTURALS    1.94     2.35
AV. TR.D. TO STRUCTURALS    0.72     0.96
MAX TR.T.                   4.84     5.02

 125 IMPROVED BOXES       CURRENT  PROPOSED ( 1068 ALARMS,  389 STRUCTS.)
AV. TR.T.                   3.42     2.67
AV. TR.D.                   1.59     1.15
AV. TR.T. TO STRUCTURALS    2.41     1.92
AV. TR.D. TO STRUCTURALS    0.99     0.70
MAX TR.T.                   5.27     4.82

 276 AFFECTED BOXES       CURRENT  PROPOSED ( 3131 ALARMS, 1210 STRUCTS.)
AV. TR.T.                   2.97     2.85
AV. TR.D.                   1.33     1.26
AV. TR.T. TO STRUCTURALS    2.09     2.21
AV. TR.D. TO STRUCTURALS    0.81     0.88
MAX TR.T.                   5.27     5.02
```
[p]

[p] Second-due engine citywide affected region statistics. Affected region statistics by demand region follow.

REGION 1

21 DEGRADED BOXES	CURRENT	PROPOSED (736 ALARMS, 293 STRUCTS.)
AV. TR.T.	1.55	2.08
AV. TR.D.	0.48	0.80
AV. TR.T. TO STRUCTURALS	1.40	1.91
AV. TR.D. TO STRUCTURALS	0.40	0.70
MAX TR.T.	2.90	3.34

17 IMPROVED BOXES	CURRENT	PROPOSED (446 ALARMS, 189 STRUCTS.)
AV. TR.T.	2.43	1.83
AV. TR.D.	1.01	0.65
AV. TR.T. TO STRUCTURALS	1.92	1.60
AV. TR.D. TO STRUCTURALS	0.71	0.51
MAX TR.T.	3.37	2.40

38 AFFECTED BOXES	CURRENT	PROPOSED (1182 ALARMS, 482 STRUCTS.)
AV. TR.T.	1.94	1.97
AV. TR.D.	0.72	0.73
AV. TR.T. TO STRUCTURALS	1.60	1.79
AV. TR.D. TO STRUCTURALS	0.52	0.63
MAX TR.T.	3.37	3.34

REGION 2

1 IMPROVED BOXES	CURRENT	PROPOSED (0 ALARMS, 0 STRUCTS.)
AV. TR.T.	4.76	3.10
AV. TR.D.	2.39	1.40
AV. TR.T. TO STRUCTURALS	0.00	0.00
AV. TR.D. TO STRUCTURALS	0.00	0.00
MAX TR.T.	4.76	3.10

REGION 3

81 DEGRADED BOXES	CURRENT	PROPOSED (788 ALARMS, 330 STRUCTS.)
AV. TR.T.	3.04	3.40
AV. TR.D.	1.37	1.58
AV. TR.T. TO STRUCTURALS	2.49	2.86
AV. TR.D. TO STRUCTURALS	1.04	1.26
MAX TR.T.	4.84	5.02

62 IMPROVED BOXES	CURRENT	PROPOSED (149 ALARMS, 45 STRUCTS.)
AV. TR.T.	3.93	3.01
AV. TR.D.	1.90	1.35
AV. TR.T. TO STRUCTURALS	2.71	2.54
AV. TR.D. TO STRUCTURALS	1.18	1.07
MAX TR.T.	5.27	4.82

143 AFFECTED BOXES	CURRENT	PROPOSED (937 ALARMS, 375 STRUCTS.)
AV. TR.T.	3.43	3.23
AV. TR.D.	1.60	1.48
AV. TR.T. TO STRUCTURALS	2.51	2.82
AV. TR.D. TO STRUCTURALS	1.06	1.24
MAX TR.T.	5.27	5.02

REGION 4

49 DEGRADED BOXES	CURRENT	PROPOSED (539 ALARMS, 198 STRUCTS.)
AV. TR.T.	2.35	2.73
AV. TR.D.	0.96	1.18
AV. TR.T. TO STRUCTURALS	1.84	2.17
AV. TR.D. TO STRUCTURALS	0.66	0.85
MAX TR.T.	3.53	3.69

45 IMPROVED BOXES	CURRENT	PROPOSED (473 ALARMS, 155 STRUCTS.)
AV. TR.T.	3.05	2.51
AV. TR.D.	1.37	1.05
AV. TR.T. TO STRUCTURALS	2.91	2.13
AV. TR.D. TO STRUCTURALS	1.29	0.83
MAX TR.T.	3.84	3.76

94 AFFECTED BOXES	CURRENT	PROPOSED (1012 ALARMS, 353 STRUCTS.)
AV. TR.T.	2.69	2.62
AV. TR.D.	1.16	1.12
AV. TR.T. TO STRUCTURALS	2.31	2.15
AV. TR.D. TO STRUCTURALS	0.94	0.84
MAX TR.T.	3.84	3.76

1ST DUE ENGINE RESPONSE

15 TARGET HAZARDS	CURRENT	PROPOSED (270 ALARMS, 102 STRUCTS.) [q]
AV. TR.T.	1.69	1.56
AV. TR.D.	0.57	0.49
AV. TR.T. TO STRUCTURALS	1.52	1.42
AV. TR.D. TO STRUCTURALS	0.47	0.41
MAX TR.T.	2.71	3.09

[q] First-due engine statistics averaged over all 15 target hazards in the city.

	BOX	TRV. CURR.	DIST. PROP.	TRV. CURR.	TIME PROP.
*	207	0.84	1.40	2.14	3.09
*	506	0.12	0.83	0.93	2.13
	612	0.23	0.23	1.12	1.12
*	707	0.33	0.41	1.28	1.43
*	804	0.58	0.41	1.70	1.43
*	807	0.51	0.26	1.60	1.17
*	809	0.74	0.16	1.97	1.01
*	810	0.51	0.36	1.60	1.35
	901	0.84	0.84	2.14	2.14
	911	0.58	0.58	1.70	1.70
*	1011	0.81	0.62	2.09	1.78
	1210	0.41	0.41	1.43	1.43
*	1416	0.62	0.00	1.78	0.73
*	1419	1.17	0.58	2.71	1.70
	1704	0.23	0.23	1.12	1.12

[r] Detailed first-due engine statistics for each target hazard. Asterisks indicate target hazards with different first-due travel times in the current and proposed configurations.

2ND DUE ENGINE RESPONSE

15 TARGET HAZARDS	CURRENT	PROPOSED (270 ALARMS, 102 STRUCTS.)
AV. TR.T.	2.49	2.44
AV. TR.D.	1.04	1.01
AV. TR.T. TO STRUCTURALS	2.25	2.16
AV. TR.D. TO STRUCTURALS	0.90	0.85
MAX TR.T.	4.03	4.29

	BOX	TRV. CURR.	DIST. PROP.	TRV. CURR.	TIME PROP.
*	207	1.65	1.14	3.51	2.65
*	506	0.93	1.04	2.30	2.49
*	612	1.27	0.99	2.86	2.40
*	707	0.67	0.84	1.86	2.14
*	804	0.41	0.81	1.43	2.10
*	807	1.20	0.49	2.76	1.56
*	809	0.77	0.65	2.03	1.83
	810	1.17	1.17	2.71	2.71
*	901	1.08	1.38	2.56	3.06
*	911	1.13	0.58	2.64	1.70
	1011	0.93	0.93	2.30	2.30
*	1210	0.41	0.77	1.43	2.03
*	1416	1.38	1.60	3.06	3.42
*	1419	1.95	2.11	4.03	4.29
	1704	0.67	0.67	1.86	1.86

1ST DUE LADDER RESPONSE [s]

CITYWIDE (398 BOXES)	CURRENT	PROPOSED (4730 ALARMS, 1872 STRUCTS.)
AV. TR.T.	2.75	2.47
AV. TR.D.	1.20	1.03
AV. TR.T. TO STRUCTURALS	1.88	1.69
AV. TR.D. TO STRUCTURALS	0.68	0.57
MAX TR.T.	5.06	5.06

REGION 1 (71 BOXES)	CURRENT	PROPOSED (2215 ALARMS, 927 STRUCTS.)
AV. TR.T.	1.76	1.76
AV. TR.D.	0.61	0.61
AV. TR.T. TO STRUCTURALS	1.54	1.54
AV. TR.D. TO STRUCTURALS	0.48	0.48
MAX TR.T.	3.10	3.10

REGION 2 (58 BOXES)	CURRENT	PROPOSED (254 ALARMS, 86 STRUCTS.)
AV. TR.T.	2.97	2.97
AV. TR.D.	1.33	1.33
AV. TR.T. TO STRUCTURALS	2.43	2.43
AV. TR.D. TO STRUCTURALS	1.01	1.01
MAX TR.T.	5.06	5.06

REGION 3 (173 BOXES)	CURRENT	PROPOSED (1211 ALARMS, 492 STRUCTS.)
AV. TR.T.	3.37	2.74
AV. TR.D.	1.56	1.19
AV. TR.T. TO STRUCTURALS	2.50	1.80
AV. TR.D. TO STRUCTURALS	1.05	0.63
MAX TR.T.	4.98	4.91

REGION 4 (96 BOXES)	CURRENT	PROPOSED (1050 ALARMS, 367 STRUCTS.)
AV. TR.T.	2.24	2.23
AV. TR.D.	0.90	0.89
AV. TR.T. TO STRUCTURALS	1.76	1.76
AV. TR.D. TO STRUCTURALS	0.61	0.61
MAX TR.T.	3.84	3.77

[s] Ladder statistics. These follow precisely the same format as the engine statistics (refer to notes *g*–*r* above).

DISTRIBUTION OF TRAVEL TIMES TO BOXES - CITYWIDE

TR TIME	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5
# BOXES		9	57	84	92	49	32	27	22	25	1

DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION 1

TR TIME	0.5	1.0	1.5	2.0	2.5	3.0	3.5
# BOXES		4	25	17	17	7	1

DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION 2

TR TIME	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0	5.5
# BOXES		3	7	17	8	4	5	5	8	1

DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION 3

TR TIME	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5	5.0
# BOXES		2	17	35	34	21	13	17	17	17

DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION 4

TR TIME	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0
# BOXES		3	12	25	24	13	14	5

2ND DUE LADDER RESPONSE

CITYWIDE (398 BOXES)	CURRENT	PROPOSED (4730 ALARMS, 1872 STRUCTS.)
AV. TR.T.	3.56	3.52
AV. TR.D.	1.68	1.65
AV. TR.T. TO STRUCTURALS	2.69	2.63
AV. TR.D. TO STRUCTURALS	1.16	1.13
MAX TR.T.	5.86	5.86

REGION 1 (71 BOXES)	CURRENT	PROPOSED (2215 ALARMS, 927 STRUCTS.)
AV. TR.T.	2.24	2.23
AV. TR.D.	0.89	0.89
AV. TR.T. TO STRUCTURALS	2.02	2.02
AV. TR.D. TO STRUCTURALS	0.77	0.76
MAX TR.T.	3.34	3.34

REGION 2 (58 BOXES)	CURRENT	PROPOSED (254 ALARMS, 86 STRUCTS.)
AV. TR.T.	3.63	3.63
AV. TR.D.	1.72	1.72
AV. TR.T. TO STRUCTURALS	3.18	3.18
AV. TR.D. TO STRUCTURALS	1.45	1.45
MAX TR.T.	5.58	5.58

REGION 3 (173 BOXES)	CURRENT	PROPOSED (1211 ALARMS, 492 STRUCTS.)
AV. TR.T.	3.90	3.82
AV. TR.D.	1.88	1.83
AV. TR.T. TO STRUCTURALS	3.36	3.15
AV. TR.D. TO STRUCTURALS	1.56	1.43
MAX TR.T.	5.86	5.86

REGION 4 (96 BOXES)	CURRENT	PROPOSED (1050 ALARMS, 367 STRUCTS.)
AV. TR.T.	3.89	3.86
AV. TR.D.	1.87	1.86
AV. TR.T. TO STRUCTURALS	3.36	3.35
AV. TR.D. TO STRUCTURALS	1.56	1.55
MAX TR.T.	5.20	5.15

```
DISTRIBUTION OF TRAVEL TIMES TO BOXES - CITYWIDE

TR TIME  1.0  1.5  2.0  2.5  3.0  3.5  4.0  4.5  5.0  5.5  6.0
# BOXES     3   27   46   52   66   62   70   40   26    6


DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  1

TR TIME  1.0  1.5  2.0  2.5  3.0  3.5
# BOXES     3   23   24   16    5


DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  2

TR TIME  2.0  2.5  3.0  3.5  4.0  4.5  5.0  5.5  6.0
# BOXES     6   11   15    6    6    6    7    1


DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  3

TR TIME  1.5  2.0  2.5  3.0  3.5  4.0  4.5  5.0  5.5  6.0
# BOXES     4   16   15   25   34   38   22   14    5


DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION  4

TR TIME  2.5  3.0  3.5  4.0  4.5  5.0  5.5
# BOXES    10   21   22   26   12    5
```

1ST DUE LADDER RESPONSE

	CO.	BOXES CURR.	BOXES PROP.	AV. TR.T. CURR.	AV. TR.T. PROP.	MAX TR.T. CURR.	MAX TR.T. PROP.	ALARMS CURR.	ALARMS PROP.	STRUCTURALS CURR.	STRUCTURALS PROP.
*	1	201	121	3.13	2.92	4.98	4.91	2134	1387	852	560
*	2	112	107	2.25	2.19	3.84	3.77	1505	1501	554	554
	3	85	85	2.51	2.51	5.06	5.06	1091	1091	466	466
*	11	0	85	0.00	2.15	0.00	4.73	0	751	0	292

2ND DUE LADDER RESPONSE

	CO.	BOXES CURR.	BOXES PROP.	AV. TR.T. CURR.	AV. TR.T. PROP.	MAX TR.T. CURR.	MAX TR.T. PROP.	ALARMS CURR.	ALARMS PROP.	STRUCTURALS CURR.	STRUCTURALS PROP.
*	1	131	119	3.50	3.41	5.58	5.58	1323	1283	521	513
*	2	11	10	2.55	2.49	3.07	2.94	241	239	118	118
*	3	256	222	3.63	3.66	5.86	5.86	3166	2794	1233	1071
*	11	0	47	0.00	3.33	0.00	5.10	0	414	0	170

1ST DUE LADDER RESPONSE

CITYWIDE

85 IMPROVED BOXES	CURRENT	PROPOSED	(751 ALARMS, 292 STRUCTS.)
AV. TR.T.	3.45	2.15	
AV. TR.D.	1.61	0.84	
AV. TR.T. TO STRUCTURALS	2.83	1.66	
AV. TR.D. TO STRUCTURALS	1.25	0.55	
MAX TR.T.	4.98	4.73	

REGION 3

80 IMPROVED BOXES	CURRENT	PROPOSED	(747 ALARMS, 292 STRUCTS.)
AV. TR.T.	3.45	2.09	
AV. TR.D.	1.61	0.80	
AV. TR.T. TO STRUCTURALS	2.83	1.66	
AV. TR.D. TO STRUCTURALS	1.25	0.55	
MAX TR.T.	4.98	4.73	

REGION 4

5 IMPROVED BOXES	CURRENT	PROPOSED	(4 ALARMS, 0 STRUCTS.)
AV. TR.T.	3.47	3.22	
AV. TR.D.	1.62	1.48	
AV. TR.T. TO STRUCTURALS	0.00	0.00	
AV. TR.D. TO STRUCTURALS	0.00	0.00	
MAX TR.T.	3.84	3.65	

2ND DUE LADDER RESPONSE

CITYWIDE

47 IMPROVED BOXES	CURRENT	PROPOSED (414 ALARMS, 170 STRUCTS.)
AV. TR.T.	3.69	3.33
AV. TR.D.	1.76	1.54
AV. TR.T. TO STRUCTURALS	2.95	2.32
AV. TR.D. TO STRUCTURALS	1.32	0.94
MAX TR.T.	5.20	5.10

REGION 1

4 IMPROVED BOXES	CURRENT	PROPOSED (41 ALARMS, 11 STRUCTS.)
AV. TR.T.	2.81	2.62
AV. TR.D.	1.23	1.12
AV. TR.T. TO STRUCTURALS	2.73	2.56
AV. TR.D. TO STRUCTURALS	1.19	1.08
MAX TR.T.	3.07	2.83

REGION 3

31 IMPROVED BOXES	CURRENT	PROPOSED (333 ALARMS, 151 STRUCTS.)
AV. TR.T.	3.54	3.09
AV. TR.D.	1.66	1.40
AV. TR.T. TO STRUCTURALS	2.90	2.21
AV. TR.D. TO STRUCTURALS	1.28	0.88
MAX TR.T.	5.15	4.91

REGION 4

12 IMPROVED BOXES	CURRENT	PROPOSED (40 ALARMS, 8 STRUCTS.)
AV. TR.T.	4.40	4.16
AV. TR.D.	2.18	2.04
AV. TR.T. TO STRUCTURALS	4.37	4.02
AV. TR.D. TO STRUCTURALS	2.16	1.95
MAX TR.T.	5.20	5.10

1ST DUE LADDER RESPONSE

15 TARGET HAZARDS	CURRENT	PROPOSED (270 ALARMS, 102 STRUCTS.)
AV. TR.T.	2.18	1.87
AV. TR.D.	0.86	0.67
AV. TR.T. TO STRUCTURALS	1.87	1.80
AV. TR.D. TO STRUCTURALS	0.67	0.63
MAX TR.T.	4.03	2.86

	BOX	TRV. CURR.	DIST. PROP.	TRV. CURR.	TIME PROP.
	207	0.84	0.84	2.14	2.14
	506	0.12	0.12	0.93	0.93
	612	1.27	1.27	2.86	2.86
	707	0.33	0.33	1.28	1.28
	804	0.58	0.58	1.70	1.70
	807	0.51	0.51	1.60	1.60
	809	0.74	0.74	1.97	1.97
	810	0.93	0.93	2.30	2.30
	901	1.08	1.08	2.56	2.56
	911	1.13	1.13	2.64	2.64
	1011	0.93	0.93	2.30	2.30
	1210	0.41	0.41	1.43	1.43
*	1416	1.38	0.00	3.06	0.73
*	1419	1.95	0.58	4.03	1.70
	1704	0.67	0.67	1.86	1.86

2ND DUE LADDER RESPONSE

15 TARGET HAZARDS	CURRENT	PROPOSED (270 ALARMS, 102 STRUCTS.)
AV. TR.T.	3.24	3.23
AV. TR.D.	1.49	1.48
AV. TR.T. TO STRUCTURALS	2.87	2.86
AV. TR.D. TO STRUCTURALS	1.27	1.26
MAX TR.T.	4.99	4.99

BOX	TRV. CURR.	DIST. PROP.	TRV. CURR.	TIME PROP.
207	2.43	2.43	4.82	4.82
506	1.73	1.73	3.64	3.64
612	1.75	1.75	3.68	3.68
707	1.40	1.40	3.09	3.09
804	1.20	1.20	2.76	2.76
807	1.17	1.17	2.71	2.71
809	1.15	1.15	2.67	2.67
810	1.17	1.17	2.71	2.71
901	1.38	1.38	3.07	3.07
911	1.18	1.18	2.73	2.73
* 1011	1.31	1.22	2.94	2.79
1210	0.88	0.88	2.21	2.21
1416	1.95	1.95	4.03	4.03
1419	2.53	2.53	4.99	4.99
1704	1.08	1.08	2.56	2.56

```
COMMAND? C 1
```
[t]

```
COMMAND? P

 1)  MOVE ENGINE 10 TO BOX 1416
     MOVE ENGINE  6 TO BOX  908

 2)  DELETE ENGINE  2
```
[u]

```
COMMAND? O C=(E),R=(A),L=Y
```
[v]

[t]The "C" command indicates that the last change command should be deleted, returning the proposed configuration to the way it was before the command was issued. See notes *c* and *e* above.

[u]The output from the "P" command shows that only the first two Move commands previously entered (see notes *a*, *b*, and *e* above) are now stored.

[v]The "O" command requests first-due engine company output for the affected region and for the city as a whole, with lists of the improved and degraded alarm boxes.

1ST DUE ENGINE RESPONSE

CITYWIDE (398 BOXES)	CURRENT	PROPOSED (4730 ALARMS, 1872 STRUCTS.)
AV. TR.T.	1.98	1.96
AV. TR.D.	0.74	0.73
AV. TR.T. TO STRUCTURALS	1.45	1.46
AV. TR.D. TO STRUCTURALS	0.42	0.43
MAX TR.T.	3.80	3.80

DISTRIBUTION OF TRAVEL TIMES TO BOXES - CITYWIDE

TR TIME	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0
# BOXES		18	97	110	85	46	35	7

$^{\omega}$First-due engine statistics for the city as a whole.

1ST DUE ENGINE RESPONSE

CITYWIDE[x]

64 DEGRADED BOXES	CURRENT	PROPOSED (1091 ALARMS, 414 STRUCTS.)
AV. TR.T.	1.57	2.23
AV. TR.D.	0.50	0.89
AV. TR.T. TO STRUCTURALS	1.32	1.92
AV. TR.D. TO STRUCTURALS	0.35	0.70
MAX TR.T.	2.39	3.43

86 IMPROVED BOXES	CURRENT	PROPOSED (1093 ALARMS, 441 STRUCTS.)
AV. TR.T.	2.27	1.69
AV. TR.D.	0.91	0.57
AV. TR.T. TO STRUCTURALS	1.88	1.37
AV. TR.D. TO STRUCTURALS	0.68	0.38
MAX TR.T.	3.55	2.73

150 AFFECTED BOXES	CURRENT	PROPOSED (2184 ALARMS, 855 STRUCTS.)
AV. TR.T.	1.97	1.92
AV. TR.D.	0.74	0.71
AV. TR.T. TO STRUCTURALS	1.61	1.63
AV. TR.D. TO STRUCTURALS	0.52	0.53
MAX TR.T.	3.55	3.43

[x] First-due engine citywide affected region statistics.

```
REGION   1

   2 DEGRADED BOXES          CURRENT   PROPOSED (    75 ALARMS,    35 STRUCTS.)
AV. TR.T.                       1.33      1.61
AV. TR.D.                       0.35      0.52
AV. TR.T. TO STRUCTURALS        1.34      1.66
AV. TR.D. TO STRUCTURALS        0.36      0.55
MAX TR.T.                       1.35      1.70

  12 IMPROVED BOXES          CURRENT   PROPOSED (   386 ALARMS,   167 STRUCTS.)
AV. TR.T.                       1.93      1.30
AV. TR.D.                       0.71      0.34
AV. TR.T. TO STRUCTURALS        1.82      1.27
AV. TR.D. TO STRUCTURALS        0.65      0.32
MAX TR.T.                       2.49      1.78

  14 AFFECTED BOXES          CURRENT   PROPOSED (   461 ALARMS,   202 STRUCTS.)
AV. TR.T.                       1.84      1.35
AV. TR.D.                       0.66      0.36
AV. TR.T. TO STRUCTURALS        1.74      1.34
AV. TR.D. TO STRUCTURALS        0.60      0.36
MAX TR.T.                       2.49      1.78

DEGRADED BOXES:
1609 1608

IMPROVED BOXES:
1110 1109 1108 1011 1010 1009 1008 1007  910  909  908  907
```
[y]

[y]Lists of box numbers of alarm boxes in Demand Region 1 whose first-due engine travel times are improved or degraded in the proposed configuration. Affected region output, including lists of affected boxes, for the other demand regions follows.

REGION 3

```
  21 DEGRADED BOXES      CURRENT  PROPOSED (  507 ALARMS,  228 STRUCTS.)
AV. TR.T.                  1.39     1.85
AV. TR.D.                  0.39     0.66
AV. TR.T. TO STRUCTURALS   1.29     1.79
AV. TR.D. TO STRUCTURALS   0.33     0.63
MAX TR.T.                  2.10     2.25

  58 IMPROVED BOXES      CURRENT  PROPOSED (  413 ALARMS,  145 STRUCTS.)
AV. TR.T.                  2.47     1.83
AV. TR.D.                  1.03     0.65
AV. TR.T. TO STRUCTURALS   2.13     1.52
AV. TR.D. TO STRUCTURALS   0.83     0.46
MAX TR.T.                  3.55     2.73

  79 AFFECTED BOXES      CURRENT  PROPOSED (  920 ALARMS,  373 STRUCTS.)
AV. TR.T.                  2.18     1.84
AV. TR.D.                  0.86     0.66
AV. TR.T. TO STRUCTURALS   1.61     1.68
AV. TR.D. TO STRUCTURALS   0.52     0.56
MAX TR.T.                  3.55     2.73

DEGRADED BOXES:
1815 1715 1714 1713 1712 1614 1613 1612 1514 1513 1512 1511 1414 1413
1412 1313 1312 1213 1212 1113 1112

IMPROVED BOXES:
2015 1915 1914 1819 1818 1816 1814 1721 1720 1719 1718 1717 1716 1621
1620 1619 1618 1617 1616 1615 1521 1520 1519 1518 1517 1516 1515 1421
1420 1419 1418 1417 1416 1415 1321 1320 1319 1318 1317 1316 1315 1314
1311 1218 1217 1216 1215 1214 1211 1116 1115 1114 1015 1014 1013 1012
 914  913
```

```
REGION   4

  41 DEGRADED BOXES       CURRENT  PROPOSED (  509 ALARMS,   151 STRUCTS.)
AV. TR.T.                   1.68      2.45
AV. TR.D.                   0.56      1.02
AV. TR.T. TO STRUCTURALS    1.36      2.18
AV. TR.D. TO STRUCTURALS    0.37      0.86
MAX TR.T.                   2.39      3.43

  16 IMPROVED BOXES       CURRENT  PROPOSED (  294 ALARMS,   129 STRUCTS.)
AV. TR.T.                   1.83      1.49
AV. TR.D.                   0.65      0.45
AV. TR.T. TO STRUCTURALS    1.66      1.33
AV. TR.D. TO STRUCTURALS    0.55      0.35
MAX TR.T.                   2.47      2.21

  57 AFFECTED BOXES       CURRENT  PROPOSED (  803 ALARMS,   280 STRUCTS.)
AV. TR.T.                   1.72      2.18
AV. TR.D.                   0.59      0.86
AV. TR.T. TO STRUCTURALS    1.50      1.79
AV. TR.D. TO STRUCTURALS    0.45      0.62
MAX TR.T.                   2.47      3.43

DEGRADED BOXES:
 707  706  705  704  703  702  608  607  606  605  604  603  602  601
 508  507  506  505  504  503  502  501  408  407  406  405  404  403
 402  401  308  307  306  305  304  207  206  205  107  106  105

IMPROVED BOXES:
 810  809  808  807  806  805  804  803  802  801  709  708  609  509
 409  309


COMMAND? E[a]
```

[a]The "E" command terminates execution.

Appendix C

FILE FORMATS

Table C.1

DATGEN INPUT FILE (DATIN)

Record Number	Column	Format	Variable	Description
1	1-2	I2	MAXPID	Maximum numeric identifier of any proposed company
	4-5	I2	MAXCID	Maximum numeric identifier of any existing company
	7-8	I2	MAXDUE(1)	Number of engine companies in each alarm assignment list
	10-11	I2	MAXDUE(2)	Number of ladder companies in each alarm assignment list
	13-14	I2	NREG	Number of demand regions in the city; this is the maximum value that can be supplied as a demand region identifier for an alarm box
	16-19	I4	NBOX	Number of alarm boxes in the city
	21	I1	RDSW	Response distance switch (see Section 7.2). If 0 or 1, compute travel distances from Euclidean distances multiplied by RDFCT If greater than or equal to 2, compute travel distances from right-angle distances with coordinate axes rotating THETA degrees
	23-25	I3	NCOMPS	Total number of companies in city

Table C.1 (continued)

Record Number	Column	Format	Variable	Description
	27	I1	KAACSW	Alarm assignment generate switch. If 0 or 1, alarm assignment lists for each box must be supplied by the user and will be used to determine travel distances and company response areas If greater than or equal to 2, DATGEN will generate alarm assignment lists for each box, with companies in order of increasing response distance
2	1-5	F5.2	RDFCT	Factor to apply to Euclidean distances to determine travel distances
	7-11	F5.2	A	Parameter A for travel time function
	13-17	F5.2	B	Parameter B for travel time function
	19-23	F5.2	C	Parameter C for travel time function
	25-29	F5.2	XDIST	Cutoff distance (d) for travel time function
	31-35	F5.2	SCALE	Number of grid units per mile
	37-41	F5.2	THETA	Angle between coordinate grid and street grid in degrees (counterclockwise positive) (see Section 7.2.)

Table C.1 (continued)

Record Number	Column	Format	Variable	Description
3 thru (NCOMPS+2)	1	A1	CTYPE	Company type; value must be 'E' or 'L', indicating an engine or ladder company
	2-3	I2	KID	Company identification number; must be right-adjusted in the field; leading zeros are permitted
	5-9	F5.1	XC	x coordinate of company's house in grid units
	11-15	F5.1	YC	y coordinate of company's house in grid units
(NCOMPS+3) thru (NCOMPS+2+NBOX) (one record for each box in the data base, in ascending order of box number)	1-4	I4	BOXNUM	Alarm box identification number
	6-10	F5.1	X	x coordinate of box in grid units
	12-16	F5.1	Y	y coordinate of box in grid units
	18	I1	HZRGN	Demand region of box
	20	I1	TRGT	Target hazard indicator (1 = yes, 0 = no)
	22-25	I4	BXTLRM	Total number of alarms at box over some time period
	27-29	I3	BXSLRM	Structural alarms at box over same time period as BXTLRM
	Starting with column 31, format is (200I2) with MAXDUE(1) entries for engine companies, followed, by MAXDUE(2) entires for ladder companies		ID(*,1)+	Numeric identifiers of engine companies that respond to the box, in the order in which they should arrive (two columns per engine

+An asterisk is used in an array reference to refer to the entire range of the corresponding subscript.

Table C.1 (continued)

Record Number	Column	Format	Variable	Description
				company); these need not be supplied if DATGEN is to generate alarm assignment lists
			ID(*,2)	Numeric identifiers of ladder companies that respond to the box, in the order in which they should arrive (two columns per ladder company); these need not be supplied if DATGEN is to generate alarm assignment lists

If more than 200 companies (engines and ladders combined) are included in the alarm assignment data, FORMAT 4 in MAIN of DATGEN must be changed.

Table C.2

SITE DATA BASE FILE (DATA)

This file is also the DATGEN output file DATOUT. It is in binary (internal) format. The dimensions of arrays are expressed in terms of the control variables defined in Table D.3. The number of elements written for each array are determined by the actual values of the control variables (not by the default maxima).

Record	Variable	Type	Description
1	MAXPID	Integer	Maximum numeric identifier for any proposed company
	MAXDUE(1)	Integer	Number of engine companies on each alarm assignment list
	MAXDUE(2)	Integer	Number of ladder companies on each alarm assignment list
	NREG	Integer	Number of demand regions in the city
	NBOX	Integer	Number of alarm boxes in the city
	NEWID	Integer	First (lowest) numeric identifier to be used for any added company (NEWID = MAXCID+1)
	RDSW	Integer	Travel distance mode switch (0 or 1 = Euclidean distance; 2 = right-angle distance)
	RDFCT	Real	Factor to apply to Euclidean distances to obtain estimated travel distances
	A	Real	Parameter A for travel time function
	B	Real	Parameter B for travel time function
	D	Real	Parameter C for travel time function
	XDIST	Real	Cutoff distance (d) for travel time function

Table C.2 (continued)

Record	Variable	Type	Description
2	BOXNUM (NBOX)	Integer	Identification numbers of alarm boxes, listed in ascending numeric order
	X(NBOX)	Real	x coordinates of alarm boxes, in the same order as they appear in BOXNUM
	Y(NBOX)	Real	y coordinates of alarm boxes, in the same order as they appear in BOXNUM
3 through NBOX + 2	HZRGN	Integer	Demand region of alarm box
(one record for each alarm	TRGT	Integer	Target hazard indicator (1 = yes; 0 = no)
box, in ascending numeric	BXTLRM	Integer	Total number of alarms at box over some time period
order)	BXSLRM	Integer	Structural alarms at box over same time period as BXTLRM
	ID(MAXDUE(1),2)	Integer	Numeric identifiers of engine and ladder companies that respond to the box. ID(I,J) is the Ith-due company of type J; J = 1 for engines, J = 2 for ladders
	RSPDST (MAXDUE(1),2)	Real	Estimated travel distances for each of the companies that respond to the box. RSPDST(I,J) is the estimated distance for the Ith-due company of type J
NBOX+3	OCRD(MAXPID, MAXDUE(1),2)	Real	Average distance of each company to the alarm boxes in their current response areas. OCRD(I,J,K) is the average travel distance for the type K company with numeric identifier = I to alarm boxes in its Jth-due response area

Table C.2 (continued)

Record	Variable	Type	Description
	OCRT(MAXPID, MAXDUE(1),2)	Real	The same as OCRD, but for average travel time
	OCMRT(MAXPID, MAXDUE(1),2)	Real	The same as OCRD, but for maximum travel time
	OCTLRM(MAXPID, MAXDUE(1),2)	Integer	The same as OCRD, but for the total number of alarms at boxes in the response area
	OCSLRM(MAXPID, MAXDUE(1),2)	Integer	The same as OCRD, but for structural alarms at boxes in the response area
	OCBOX(MAXPID, MAXDUE(1),2)	Integer	The same as OCRD, but for the number of alarm boxes in the response area
NBOX+4	ORRD(NREG+2, MAXDUE(1),2)	Real	Current average travel distance to alarm boxes in each region. ORRD(I,J,K) is the average Jth-due travel distance of companies of type K in region I. I = 1 is for the artificial region composed of all boxes at which there are target hazards; I = 2 is for the entire city; I = 3 through NREG+2 are for the demand regions in order of region number
	ORWRD(NREG+2, MAXDUE(1),2)	Real	The same as ORRD, but for travel distance weighted by the total number of alarms at each alarm box in the region

Table C.2 (continued)

Record	Variable	Type	Description
	ORRT(NREG+2, MAXDUE(1),2)	Real	The same as ORRD, but for average travel time in each of the regions
	ORWRT(NREG+2, MAXDUE(1),2)	Real	The same as ORRD, but for travel time weighted by the the total number of alarms at each alarm box in the region
	ORMRT(NREG+2, MAXDUE(1),2)	Real	The same as ORRD, but for maximum travel time
	ORSWRD(NREG+2, MAXDUE(1),2)	Real	The same as ORRD, but for travel distance weighted by structural alarms at each alarm box in the region
	ORSWRT(NREG+2, MAXDUE(1),2)	Real	The same as ORRD, but for travel time weighted by structural alarms at each alarm box in the region
	RGNBOX(NREG+2)	Integer	The number of alarm boxes in each of the regions
	RGTLRM(NREG+2)	Integer	The number of alarms in each of the regions
	RGSLRM(NREG+2)	Integer	The number of structural fires in each of the regions

Table C.3

SITE MAPPING OUTPUT FILE (MAP)

This file is produced by the SITE program when the user specifies "M = Y" in an "O" command. The file contains sufficient information to produce shaded maps of travel times and incident densities using the GRIDS mapping system (see Section 4.4). The file contains one record for each alarm box. The records include engine and ladder travel times up to and including the response level specified in the "O" command. Therefore, the actual length of the records created depends upon the response level specified. First-due travel times are always included. The format of the record for each alarm box is:

Column	Format	Description
1	I4	Numeric identifier of the alarm box
5	F8.5	x coordinate
13	F8.5	y coordinate
21	I2	Demand region
23	I4	Total alarm count
27	I4	Structural alarm count
31	F5.2	First-due engine travel time
36	I3	First-due engine travel time change indicator (0 = no change; 1 = improved; -1 = degraded)
39	F5.2	First-due ladder travel time
44	I3	First-due ladder travel time change indicator
47	F5.2	Second-due engine travel time (if specified)
52	I3	Second-due engine travel time change indicator (if specified)
55	F5.2	Second-due ladder travel time (if specified)
.	.	.
.	.	.
.	.	.

Appendix D

TABLES FOR PROGRAM INSTALLATION

Table D.1

DATGEN FILES

File (Variable) Name	Device	Use	Default Unit Number	Mode
SYSOUT	Printer or TTY*	Print control and diagnostic information	5	character
SYSIN	Card reader or TTY*	Go/no go indicator	4	character
DATIN	Tape, direct access, cards	Input control and box data	30	character
DATOUT	Tape, direct access	Data base output (control, box, and summary data)	31	binary
SCRTCH	Tape, direct access	Scratch file	32	binary

*Teletype or other terminal device.

Table D.2

SITE FILES

File (Variable) Name	Device	Use	Default Unit Number	Mode
IUNIT (to change default, change value of DATA in COMMON/SYSTEM/)	Tape, direct access	Data base input (control, incident, and summary data)	31	binary
SYSIN	TTY, cards	User-command input	4	character
SYSOUT	Printer, TTY	Program-printed output (statistics, prompts, diagnostics)	5	character
MAP	Tape, cards, direct access	Output for mapping (box coordinate, response time, and incident data)	32	character
AAC	Printer, tape, cards, direct access	Alarm assignment output	33	character
OUNIT	Tape, direct access	Data base output (from proposed configuration)	0	binary

Table D.3

CONTROL VARIABLES

Variable	Description	Default Maximum
NBOX	Number of alarm boxes in the city	750
NREG	Number of demand regions in the city	6
MAXDUE	Number of engine companies and ladder companies included in alarm assignment data. This is a single subscript array with 2 elements: MAXDUE(1) for engines and MAXDUE(2) for ladders	10, 5
MAXPID	Maximum value of company suffix allowed for proposed companies (i.e., the maximum value of nn for Enn or Lnn)	30
MAXCID	Maximum value of company suffix allowed for current companies	29
NTGTS	Number of target hazards	25

Table D.4

ARRAYS IN THE DATA BASE PRODUCED BY DATGEN AND USED BY SITE

Array Name	Contents	Type	Minimum Dimension(s)	Occurrence in DATGEN	Occurrence in SITE
BOXNUM	List of numerical identifiers for alarm boxes (in ascending order)	Integer	(NBOX)	MAIN	COMMON/LISTS/
X	x coordinates of alarm boxes (on a grid where 1 unit = 1 mile)	Real	(NBOX)	MAIN	COMMON/LISTS/
Y	y coordinates of alarm boxes (on a grid where 1 unit = 1 mile)	Real	(NBOX)	MAIN	COMMON/LISTS/
ID	Alarm assignment data for an alarm box (company identification numbers)	Integer	(MAXPID,2)	MAIN	COMMON/BOX/
RSPDST	Travel distance data for an alarm box (entries correspond to the companies in the ID lists)	Real	(MAXPID,2)	MAIN	COMMON/BOX/
OCRT	Average travel times in current response areas of companies	Real	(MAXPID,MAXDUE(1),2)	MAIN	COMMON/COMP/
OCMRT	Maximum travel times in current response areas of companies	Real	(MAXPID,MAXDUE(1),2)	MAIN	COMMON/COMP/
OCTLRM	Total number of alarms in current response areas of companies	Integer	(MAXPID,MAXDUE(1),2)	MAIN	COMMON/COMP/
OCSLRM	Structural alarms in current response areas of companies	Integer	(MAXPID,MAXDUE(1),2)	MAIN	COMMON/COMP/

Table D.4 (Continued)

Name	Contents	Type	Minimum Dimensions	Occurrence in DATGEN	Occurrence in SITE
OCBOX	Number of alarm boxes in current response areas of companies	Integer	(MAXPID,MAXDUE(1),2)	MAIN	COMMON/COMP/
ORRD	Current average travel distances in various regions	Real	((NREG+1)*3+1,[+] MAXDUE(1),2)	COMMON POST (MAIN, RPOST, BLOCK DATA)	COMMON/OLDREG/
ORWRD	Current average travel distances in various regions, weighted by total number of alarms	Real	"	"	"
ORRT	Current average travel times in various regions	"	"	"	"
ORWRT	Current average travel times in various regions, weighted by total number of alarms	"	"	"	"
ORMRT	Current maximum travel times in various regions	"	"	"	"
ORSWRD	Current average travel distances in various regions, weighted by structural alarms	"	(NREG+2,MAXDUE(1),2)	"	"
ORSWRT	Current average travel times in various regions, weighted by structural alarms	"	"	"	"
RGNBOX	Number of alarm boxes in various regions	Integer	(NREG+2)	MAIN	"

[+] The planes of arrays of region statistics that are dimensioned in this form are allocated as follows: (1,*,*) is used for hazards; (2,*,*) for the citywide region; (3:NREG+2,*,*) for demand regions; (NREG+3,*,*) for citywide degraded boxes; (NREG+4,*,*) for citywide improved boxes; (NREG+5,*,*) for Demand Region 1 degraded boxes; (NREG+6,*,*) for Demand Region 1 improved boxes, etc.

Table D.4 (Continued)

Name	Contents	Type	Minimum Dimensions	Occurrence in DATGEN	Occurrence in SITE
RGTLRM	Total number of alarms in various regions	Integer	(NREG+2)	MAIN	COMMON/OLDREG/
RGSLRM	Number of structural alarms in various regions	"	"	"	"

Table D.5

CITY-DEPENDENT ARRAYS IN SITE

Name	Contents	Type	Minimum Dimensions	Occurrences
NRRD	Average travel distances for various regions in proposed configuration	Real	((NREG+1)*3+1,MAXDUE(1),2)	COMMON/NEWREG/
NRWRD	Average travel distances for various regions in proposed configuration, weighted by total number of alarms	Real	"	"
NRRT	Average travel times for various regions in proposed configuration	Real	"	"
NRWRT	Average travel times for various regions in proposed configuration, weighted by total number of alarms	Real	"	"
NRMRT	Maximum travel times for various regions in proposed configuration	Real	"	"
NRSWRD	Average travel distances for various regions in proposed configuration, weighted by structural alarms	Real	(NREG+2,MAXDUE(1),2)	"
NRSWRT	Average travel times for various regions in proposed configuration, weighted by structural alarms	Real	"	"
AFNBOX	Number of alarm boxes in the affected region	Integer	((NREG+1)*2,MAXDUE(1),2)	COMMON/OLDREG/

Table D.5 (Continued)

Name	Contents	Type	Minimum Dimensions	Occurrences
AFTLRM	Total number of alarms at alarm boxes in the affected region	Integer	((NREG+1)*2,MAXDUE(1),2)	COMMON/OLDREG/
AFSLRM	Number of structural alarms at alarm boxes in the affected region	Integer	"	"
RHIST	Histograms of travel times to alarm boxes in various regions	Integer	(31,NREG+1,MAXDUE(1),2)	COMMON/NEWREG/
ABXPTR	Headers for lists of affected alarm boxes	Integer	(NREG*2,MAXDUE(1),2)	"
CBXPTR	Headers for lists of alarm boxes in company response areas	Integer	(MAXPID,MAXDUE(1),2)	COMMON/COMP/
BXLST	Linked lists of alarm boxes	Integer	(NBOX,NLISTS) (NLISTS defined in Section 7.5)	COMMON/LISTS/
LIST	Pointers to storage areas of linked lists within BXLST	Integer	(2,MAXDUE(1),2)	"
NCRT	Average travel times in response areas of companies under proposed configuration	Real	(MAXPID,MAXDUE(1),2)	COMMON/COMP/
NCMRT	Maximum response times in response areas of companies under proposed configuration	Real	"	"
NCTLRM	Total number of alarms in response areas of companies under proposed configuration	Integer	"	"

Table D.5 (Continued)

Name	Contents	Type	Minimum Dimensions	Occurrences
NCSLRM	Structural alarms in response areas of companies under proposed configuration	Integer	(MAXPID,MAXDUE(1),2)	COMMON/COMP/
NCBOX	Number of alarm boxes in response areas of companies under proposed configuration	Integer	"	"
TGBXN	Internal identifiers for alarm boxes that are target hazards	Integer	(NTGTS)	MAIN
OTGRD	Current travel distances to target hazards	Real	(NTGTS,MAXDUE(1),2)	"
OTGRT	Current travel times to target hazards	Real	"	"
NTGRD	Travel distances to target hazards under proposed configuration	Real	(NTGTS,MAXDUE(1),2)	"
NTGRT	Travel times to target hazards under proposed configuration	Real	"	"
TRD	Temporary storage for RSPDST	Real	(MAXDUE(1),2)	COMMON/BOX/
RTIME	Travel time data for each alarm box	Real	"	"
CHANGE	Affected region indicators	Integer	"	"

Table D.6

STORAGE AVAILABILITY-DEPENDENT ARRAYS

Name	Dimensions	Occurrences
BXLST	(NBOX,NLISTS)	COMMON/LISTS/
BETACD	(BETAMX)	COMMON/CV/

Appendix E

PROGRAM LISTINGS

DATGEN Program

```
DATGEN.FOR[4016,10]      13:21    23-MAY-75        PAGE 1


C NEW YORK CITY - RAND INSTITUTE
C FIRE HOUSE SITE EVALUATION MODEL
C
C   DATA BASE GENERATION PROGRAM (DATGEN)
C
C
C REGION STATS
          COMMON/POST/RD,RT,BXTLRM,BXSLRM,ORRD(8,10,2),ORWRD(8,10,2),
     1ORRT(8,10,2),ORWRT(8,10,2),ORMRT(8,10,2),ORSWRD(8,10,2),
     1ORSWRT(8,10,2)
C
C CITY DEPENDENT CONTROL INFORMATION
          COMMON/CITY/MAXPID,MAXCID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,
     1 C,XDIST,SCALE,RDSW
C
C SYSTEM DEPENDENT PARAMETERS
          COMMON/SYSTEM/DATIN,DATOUT,SYSOUT,SCRTCH,SYSIN
      INTEGER DATIN,DATOUT,SYSOUT,SCRTCH,SYSIN
C
C BOX DATA
      DIMENSION X(750),Y(750),BOXNUM(750)
      DIMENSION ID(30,2),RSPDST(30,2)
C COMPANY DATA
      DIMENSION CX(30,2),CY(30,2)
        DATA CX/60*1E11/,CY/60*1E11/
C
      DIMENSION OCRT(30,10,2),OCMRT(30,10,2),
     1OCTLRM(30,10,2),OCSLRM(30,10,2),OCBOX(30,10,2)
C
      DATA OCRT/600*0./,OCMRT/600*0./,OCTLRM/600*0/,
     1OCSLRM/600*0/,OCBOX/600*0/
C
      INTEGER RGNBOX,RGTLRM,RGSLRM,BOXNUM,OCTLRM,OCSLRM,OCBOX,HZRGN,
     1BXTLRM,BXSLRM,DATIN,DATOUT,SYSOUT,TRGT,RDSW,TRGNS
C
C REGION DATA
      DIMENSION RGNBOX(8),RGTLRM(8),RGSLRM(8)
      DATA RGNBOX/8*0/,RGTLRM/8*0/,RGSLRM/8*0/
C
        DATA CHARE/1HE/,CHARL/1HL/,CHARY/1HY/
C
        DATA LSTBX/0/
C
        DIMENSION CNAME(2)
        DOUBLE PRECISION CNAME
        DATA CNAME/6HENGINE,6HLADDER/
C GET CONTROL INFO FROM DATIN
         READ (DATIN,7) MAXPID,MAXCID,MAXDUE,NREG,NBOX,RDSW,NCOMPS,
```

```
     1KAACSW,RDFCT,A,B,C,XDIST,SCALE,THETA
        IF (RDFCT .LE. 0.) RDFCT=1.15
        IF(SCALE .LE. 0.) SCALE=1.
        IF(A+B+C+XDIST .GT. 0.) GO TO 5
        A=.65
        B=1.7
        C=2.1
        XDIST=.38
C WRITE HEADING ON SYSOUT
5         WRITE(SYSOUT,8) NBOX,NREG,MAXDUE,MAXCID,MAXPID,NCOMPS,A,B,C,
     1 XDIST,SCALE
        IF (RDSW .EQ. 2) GO TO 10
        WRITE(SYSOUT,9) RDFCT
        GO TO 20
10        WRITE (SYSOUT,11) THETA
20      CONTINUE
        IF(KAACSW .GT. 1) WRITE(SYSOUT,12)
        IF(KAACSW .LE. 1) WRITE(SYSOUT,13)
        IF(MAXDUE(1) .GE. MAXDUE(2)) GO TO 22
        WRITE(SYSOUT,9023)
        STOP
22      WRITE(SYSOUT,14)
        READ(SYSIN,3) SIGNAL
        IF(SIGNAL .NE. CHARY) STOP
C GET 1ST ID OF ADDED COMPANIES
         NEWID=MAXCID+1
         MXD=MAXDUE(1)
         MXD2=MAXDUE(2)
C GET AXIS ROTATION ANGLE IN RADIANS
       IF (THETA .NE. 0. .AND. RDSW .EQ. 2) GO TO 111
C GET COMPANY COORDINATES
        DO 50 I=1,NCOMPS
        READ(DATIN,3) CTYPE,KID,XC,YC
        J=0
        IF(CTYPE .EQ. CHARE) J=1
        IF(CTYPE .EQ. CHARL) J=2
        IF(J .NE. 0 .AND. KID .GT. 0 .AND. KID .LE. MAXCID) GO TO 25
        WRITE(SYSOUT,15) CTYPE,KID,XC,YC
        STOP
25      CX(KID,J)=XC
        CY(KID,J)=YC
50      CONTINUE
C READ BOX DATA; SAVE ON FILE SCRTCH
        DO 100 I=1,NBOX
        IF(KAACSW .GT. 1) READ(DATIN,4) BOXNUM(I),X(I),Y(I),HZRGN,TRGT,
     1 BXTLRM,BXSLRM
        IF(KAACSW .LT. 2)
     1  READ(DATIN,4) BOXNUM(I),X(I),Y(I),HZRGN,TRGT,BXTLRM,BXSLRM,
```

```
     1 (ID(J,1),J=1,MXD),(ID(J,2),J=1,MXD2)
        IF(BOXNUM(I) .GT. LSTBX) GO TO 90
        WRITE(SYSOUT,19) BOXNUM(I)
        STOP
90      LSTBX=BOXNUM(I)
        IF(HZRGN .GT. 0 .AND. HZRGN .LE. NREG) GO TO 100
        WRITE(SYSOUT,21) BOXNUM(I),HZRGN
        STOP
100     WRITE(SCRTCH) HZRGN,TRGT,BXTLRM,BXSLRM,ID
        REWIND SCRTCH
C CONVERT BOX COORDS TO MILES
      DO 105 I=1,NBOX
        X(I)=X(I)/SCALE
105     Y(I)=Y(I)/SCALE
C CONVERT COMPANY COORDS TO MILES
      DO 110 J=1,2
      DO 110 I=1,MAXCID
       IF(CX(I,J) .GT. 1E10) GO TO 110
        CX(I,J)=CX(I,J)/SCALE
        CY(I,J)=CY(I,J)/SCALE
110    CONTINUE
      GO TO 114
C ROTATE COORDINATE AXES AND CONVERT TO MILES
111   DO 112 I=1,NBOX
        TX=X(I)
        X(I)=(TX*COS(THETA)+Y(I)*SIN(THETA))/SCALE
112     Y(I)=(Y(I)*COS(THETA) - TX*SIN(THETA))/SCALE
      DO 113 J=1,2
      DO 113 I=1,MAXCID
        TX=CX(I,J)
        CX(I,J)=(TX*COS(THETA)+CY(I,J)*SIN(THETA))/SCALE
113     CY(I,J)=(CY(I,J)*COS(THETA)-TX*SIN(THETA))/SCALE
C WRITE CONTROL INFO IN DATA BASE
114   WRITE(DATOUT) MAXPID,MAXDUE,NREG,NBOX,NEWID,RDSW,RDFCT,A,B,C,XDIST
        WRITE(DATOUT) (BOXNUM(I),X(I),Y(I),I=1,NBOX)
C
C PROCESS BOX RECORDS
C
      DO 199 IBX=1,NBOX
C READ DEMAND REGION, TARGET HAZARD FLAG, ALARMS, STRUCTURALS,
C     ENGINE ID'S, LADDER ID'S,
        READ(SCRTCH) HZRGN,TRGT,BXTLRM,BXSLRM,ID
C INCREMENT CITY COUNTS
         RGNBOX(2)=RGNBOX(2)+1
         RGTLRM(2)=RGTLRM(2)+BXTLRM
         RGSLRM(2)=RGSLRM(2)+BXSLRM
         I=HZRGN+2
C INCREMENT DEMAND REGION COUNTS
```

```
          RGNBOX(I)=RGNBOX(I)+1
          RGTLRM(I)=RGTLRM(I)+BXTLRM
          RGSLRM(I)=RGSLRM(I)+BXSLRM
        IF(TRGT .EQ. 0) GO TO 115
C INCREMENT TARGET HAZARD COUNTS IF REQUIRED
          RGNBOX(1)=RGNBOX(1)+1
          RGTLRM(1)=RGTLRM(1)+BXTLRM
          RGSLRM(1)=RGSLRM(1)+BXSLRM
115     CONTINUE
C POST STATS FOR BOTH COMPANY TYPES
        DO 197 K=1,2
         IMX=MAXDUE(K)
         IF(IMX .LT. 1) GO TO 197
         IERR=0
         IF(KAACSW .LT. 2) GO TO 116
         CALL AACGEN(ID(1,K),RSPDST(1,K),X(IBX),Y(IBX),CX(1,K),CY(1,K))
         GO TO 117
116      DO 1164 J=1,IMX
         IID=ID(J,K)
         IF(IID .GT. 0 .AND. IID .LE. MAXCID .AND. CX(IID,K) .LE. 1E10)
     1   GO TO 1164
         IF(IERR .EQ. 0) GO TO 1163
         WRITE(SYSOUT,16)
         STOP
1163     WRITE(SYSOUT,17) CNAME(K),BOXNUM(IBX),(ID(J,K),J=1,IMX)
         CALL AACGEN(ID(1,K),RSPDST(1,K),X(IBX),Y(IBX),CX(1,K),CY(1,K))
         IERR=1
         GO TO 116
1164     CONTINUE
117      DO 195 J=1,IMX
         IID=ID(J,K)
119      IF(KAACSW .LT. 2 .AND. IERR .EQ. 0) GO TO 120
         RD=RSPDST(J,K)
         GO TO 126
C COMPUTE TRAVEL DISTANVE BY EUCLIDIAN OR RIGHT ANGLE METHOD
120     IF (RDSW .EQ. 2) GO TO 122
          RD=SQRT((X(IBX)-CX(IID,K))**2+(Y(IBX)-CY(IID,K))**2)*RDFCT
        GO TO 125
122       RD=ABS(X(IBX)-CX(IID,K))+ABS(Y(IBX)-CY(IID,K))
C SET TRAVEL DISTANCE FOR OUTPUT
125       RSPDST(J,K)=RD
C GET TRAVEL TIME
126            RT=RSPTM(RD)
C POST CITYWIDE STATS
           CALL RPOST(2,J,K)
C POST TARGET HAZARD STATS IF REQUIRED
           IF (TRGT .EQ. 1) CALL RPOST(1,J,K)
C POST DEMAND REGION STATS
```

```
          CALL RPOST(HZRGN+2,J,K)
C POST COMPANY RESPONSE AREA STATS
         OCRT(IID,J,K)=OCRT(IID,J,K)+RT
        IF(OCMRT(IID,J,K) .LT. RT) OCMRT(IID,J,K)=RT
         OCTLRM(IID,J,K)=OCTLRM(IID,J,K)+BXTLRM
         OCSLRM(IID,J,K)=OCSLRM(IID,J,K)+BXSLRM
         OCBOX(IID,J,K)=OCBOX(IID,J,K)+1
195     CONTINUE
         IF(IERR .NE. 0) WRITE(SYSOUT,6) (ID(J,K),J=1,IMX)
197     CONTINUE
C WRITE BOX RECORD IN DATA BASE
         WRITE(DATOUT) HZRGN,TRGT,BXTLRM,BXSLRM,(((ID(J,K),RSPDST(J,K)),
     1  J=1,MXD),K=1,2)
199     CONTINUE
C
C COMPUTE SUMMARY STATISTICS
C
        DO 299 K=1,2
          IMX=MAXDUE(K)
        DO 297 J=1,IMX
C COMPUTE AVERAGE COMPANY TRAVEL TIMES
        DO 210 I=1,MAXCID
          CBX=OCBOX(I,J,K)
        IF(CBX .EQ. 0.) GO TO 210
          OCRT(I,J,K)=OCRT(I,J,K)/CBX
210     CONTINUE
297     CONTINUE
299     CONTINUE
          TRGNS=NREG+2
C WRITE SUMMARY STATS IN DATA BASE
C
         WRITE (DATOUT)
     1  (((OCRT(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1  (((OCMRT(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1  (((OCTLRM(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1  (((OCSLRM(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1  (((OCBOX(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2)
         WRITE(DATOUT)
     1  (((ORRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORWRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORWRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORMRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORSWRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORSWRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (RGNBOX(I),RGTLRM(I),RGSLRM(I),I=1,TRGNS)
         WRITE(SYSOUT,18)
          STOP
```

```
7       FORMAT(5(I2,1X),I4,1X,I1,1X,I3,1X,I1/7(F5.2,1X))
8       FORMAT(25X,'NEW YORK CITY - RAND INSTITUTE'/24X,
     1 'FIRE HOUSE SITE EVALUATION MODEL'//29X,'DATA BASE GENERATION'//
     1 1X,'DATA BASE PARAMETERS'//' NUMBER OF BOXES: ',I4/' NUMBER OF ',
     1'DEMAND REGIONS: ',
     1 I2/' MAXIMUM # ENGINES ON ALARM ASSIGNMENT CARD: ',I2/
     1 ' MAXIMUM # LADDERS ON ALARM ASSIGNMENT CARD: ',I2/
     1 ' MAXIMUM ID OF EXISTING COMPANIES: ',I2/
     1 ' MAXIMUM ID OF COMPANIES TO BE ADDED: ',I2/
     1 ' NUMBER OF COMPANIES IN CITY: ',I3/
     1 ' RESPONSE TIME SPLINE FUNCTION PARAMETERS: A = ',F5.3,', B = ',
     1F5.3, ', C = ',F5.3/43X,'CUTOFF = ',F5.3/
     1 ' SCALE: ',F4.1,' COORDINATE UNITS/MILE')
9       FORMAT(' RESPONSE DISTANCE DETERMINED FROM EUCLIDIAN DISTANCE WITH
     1 FACTOR ',F4.2)
11      FORMAT(' RESPONSE DISTANCE DETERMINED FROM RIGHT ANGLE DISTANCE WI
     1TH COORDINATE AXES ROTATED ',F6.2,' DEGREES')
3          FORMAT(A1,I2,2(1X,F5.1))
4          FORMAT(I4,2(1X,F5.1),2(1X,I1),1X,I4,1X,I3,1X,200I2)
17         FORMAT('0*** INVALID ',A6,' COMPANY FOR BOX ',I4/
     1 5X,'CURRENT ASSIGNMENT:    ',15(1X,I2)/(26X,15(1X,I2)))
6          FORMAT(5X,'CORRECTED ASSIGNMENT: ',15(1X,I2)/(26X,15(1X,I2)))
12         FORMAT(' ALARM ASSIGNMENT WILL BE GENERATED.')
13         FORMAT(' ALARM ASSIGNMENT LISTS ARE USER-SUPPLIED.')
14         FORMAT('0CONTINUE? ',$)
15         FORMAT('0*** INVALID COMPANY ID: ',A1,I2,2(1X,F5.1),
     1 ' - EXECUTION TERMINATED')
16         FORMAT('0*** INTERNAL ERROR: FAILED TO CORRECT ALARM ASSIGNMENT'/
     1        '                EXECUTION TERMINATED')
19         FORMAT('0*** BOX ',I4,' OUT OF ORDER - EXECUTION TERMINATED')
18         FORMAT('0DATA BASE GENERATION SUCCESSFUL')
21         FORMAT('0*** INVALID DEMAND REGION FOR BOX ',I4,': ',I1,
     1 ' - EXECUTION TERMINATED')
9023       FORMAT('0 *** MORE LADDERS THAN ENGINES IN ALARM ASSIGNMENT -',
     1 ' EXECUTION TERMINATED')
           END
       BLOCK DATA
           COMMON/POST/RD,RT,BXTLRM,BXSLRM,ORRD(8,10,2),ORWRD(8,10,2),
     1ORRT(8,10,2),ORWRT(8,10,2),ORMRT(8,10,2),ORSWRD(8,10,2),
     1ORSWRT(8,10,2)
C
           COMMON/CITY/MAXPID,MAXCID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,
     1 C,XDIST,SCALE,RDSW
C
           COMMON/SYSTEM/DATIN,DATOUT,SYSOUT,SCRTCH,SYSIN
       INTEGER DATIN,DATOUT,SYSOUT,SCRTCH,SYSIN
C
       DATA DATIN/30/,DATOUT/31/,SYSOUT/5/,SCRTCH/32/,SYSIN/4/
```

```
       DATA ORRD/160*0.,/,ORWRD/160*0.,/,ORRT/160*0.,/,ORWRT/160*0.,/,ORMRT/
      1 160*0.,/,ORSWRD/160*0.,/,ORSWRT/160*0.,/
          END
C
C SPLINE FUNCTION TO DETERMINE TRAVEL TIME FROM TRAVEL DISTANCE
C
          FUNCTION RSPTM(RD)
C
          COMMON/CITY/MAXPID,MAXCID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,
      1 C,XDIST,SCALE,RDSW
C
        IF (RD .GT. XDIST) GO TO 100
          RSPTM=C*SQRT(RD)
          RETURN
100       RSPTM=A+B*RD
          RETURN
          END
C
C ROUTINE TO POST REGION 'I','J'TH DUE, COMPANY TYPE 'K' SUMS
C
          SUBROUTINE RPOST(I,J,K)
C
          COMMON/POST/RD,RT,BXTLRM,BXSLRM,ORRD(8,10,2),ORWRD(8,10,2),
      1ORRT(8,10,2),ORWRT(8,10,2),ORMRT(8,10,2),ORSWRD(8,10,2),
      1ORSWRT(8,10,2)
C
       INTEGER BXTLRM,BXSLRM
C
          ORRD(I,J,K)=ORRD(I,J,K)+RD
          ORWRD(I,J,K)=ORWRD(I,J,K)+RD*BXTLRM
          ORRT(I,J,K)=ORRT(I,J,K)+RT
          ORWRT(I,J,K)=ORWRT(I,J,K)+RT*BXTLRM
          ORSWRD(I,J,K)=ORSWRD(I,J,K)+RD*BXSLRM
          ORSWRT(I,J,K)=ORSWRT(I,J,K)+RT*BXSLRM
        IF(RT .GT. ORMRT(I,J,K)) ORMRT(I,J,K)=RT
          RETURN
          END
C
C SUBROUTINE TO GENERATE ALARM ASSIGNMENT LISTS
C
         SUBROUTINE AACGEN(ID,RD,X,Y,CX,CY)
          COMMON/CITY/MAXPID,MAXCID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,
      1 C,XDIST,SCALE,RDSW
         INTEGER RDSW
         DIMENSION ID(1),RD(1),CX(1),CY(1)
C
         N=0
C COMPUTE RESPONSE DISTANCES FOR ALL COMPANIES OF SELECTED TYPE
```

```
      DO 50 I=1,MAXCID
      IF(CX(I) .GT. 1E10 .AND. CY(I) .GT. 1E10) GO TO 50
      N=N+1
      ID(N)=I
      IF(RDSW .GT. 1) GO TO 30
      RD(N)=SQRT((X-CX(I))**2+(Y-CY(I))**2)*RDFCT
      GO TO 50
30    RD(N)=ABS(X-CX(I))+ABS(Y-CY(I))
50    CONTINUE
C SORT COMPANIES BY RESPONSE DISTANCE
      NM1=N-1
      DO 100 J=1,NM1
      RDT=RD(J+1)
      IDT=ID(J+1)
      IL=0
      IU=J+1
55    IF(IU-IL .LE. 1) GO TO 70
      I=(IU+IL)/2
      IF(RDT .LT. RD(I)) GO TO 60
      IL=I
      GO TO 55
60    IU=I
      GO TO 55
70    IF(J .LT. IU) GO TO 100
      ICT=J+1
      DO 80 I=IU,J
      ICTM1=ICT-1
      RD(ICT)=RD(ICTM1)
      ID(ICT)=ID(ICTM1)
80    ICT=ICTM1
      RD(IU)=RDT
      ID(IU)=IDT
100   CONTINUE
      RETURN
      END
```

SITE Program

```
SITE.FOR[4016,10]        13:23   23-MAY-75        PAGE 1

C                  NEW YORK CITY - RAND INSTITUTE
C                 FIRE HOUSE SITE EVALUATION MODEL
C
C
C READER/INTERPRETER CONSTANTS
        COMMON/CODES/WORD,NUM,EQ,LP,RP,STR,END,CMNDS(7),TEMPL(4,2)
C SYSTEM PARAMETERS
        COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
        INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C CITY PARAMETERS
        COMMON/CITY/MAXPID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,C,XDIST,
     1 TNEWID(2),RDSW
        INTEGER TNEWID,RDSW
C COMMUNICATIONS VECTOR
        COMMON/CV/IC,BETAMX,BETACD(200)
C
        INTEGER BETAMX,BETACD
C PROPOSED REGION STATISTICS
        COMMON/NEWREG/RD,RT,AWRD,AWRT,SWRD,SWRT,RHIST(31,7,10,2)
C
        DIMENSION NRRD(22,10,2),NRWRD(22,10,2),NRRT(22,10,2),
     1 NRWRT(22,10,2),NRMRT(22,10,2),NRSWRD(8,10,2),NRSWRT(8,10,2)
C
        EQUIVALENCE (NRRD,RHIST(1,1,6,2)),(NRWRD,NRRD(1,6,2)),
     1 (NRRT,NRWRD(1,6,2)),(NRWRT,NRRT(1,6,2)),(NRMRT,NRWRT(1,6,2)),
     1 (NRSWRD,NRMRT(1,6,2)),(NRSWRT,NRSWRD(1,6,2))
C
        REAL NRRD,NRWRD,NRRT,NRWRT,NRMRT,NRSWRT,NRSWRD
        INTEGER RHIST
C CURRENT REGION STATISTICS
C
        COMMON/OLDREG/RGNBOX(8),RGTLRM(8),RGSLRM(8),AFNBOX(14,10,2)
C
        DIMENSION ORWRD(22,10,2),ORRT(22,10,2),ORWRT(22,10,2),
     1 ORMRT(22,10,2),ORSWRD(8,10,2),ORSWRT(8,10,2),
     1 ORRD(22,10,2),AFTLRM(14,10,2),AFSLRM(14,10,2),
     1 ABXPTR(14,10,2)
C
        EQUIVALENCE (AFTLRM,AFNBOX(1,6,2)),(AFSLRM,AFTLRM(1,6,2)),
     1 (ABXPTR,AFSLRM(1,6,2)),(ORRD,ABXPTR(1,6,2)),
     1 (ORWRD,ORRD(1,6,2)),(ORRT,ORWRD(1,6,2)),
     1 (ORWRT,ORRT(1,6,2)),(ORMRT,ORWRT(1,6,2)),
     1 (ORSWRD,ORMRT(1,6,2)),(ORSWRT,ORSWRD(1,6,2))
C
        INTEGER RGNBOX,RGTLRM,RGSLRM,AFNBOX,AFTLRM,AFSLRM,ABXPTR
C
       INTEGER WORD,EQ,RP,STR,END,CMNDS,TEMPL
C
```

```
C
C
C
        COMMON/LISTS/NLISTS,LIST(2,10,2),BOXNUM(750),BXLST(750,10)
C
        DIMENSION X(750),Y(750)
C
        EQUIVALENCE (X,BXLST(1,1)),(Y,BXLST(1,2))
C
        INTEGER BOXNUM,BXLST
C
        COMMON/COMP/NCRT(30,10,2)
C
      DIMENSION OCRT(30,10,2),OCMRT(30,10,2),OCTLRM(30,10,2),
     1OCSLRM(30,10,2),OCBOX(30,10,2),
     1NCMRT(30,10,2),NCTLRM(30,10,2),NCSLRM(30,10,2),NCBOX(30,10,2),
     1CBXPTR(30,10,2)
C
      EQUIVALENCE (NCMRT,NCRT(1,6,2)),(NCTLRM,NCMRT(1,6,2)),
     1(NCSLRM,NCTLRM(1,6,2)),(NCBOX,NCSLRM(1,6,2)),(CBXPTR,NCBOX(1,6,2))
     1,(OCRT,CBXPTR(1,6,2)),(OCMRT,OCRT(1,6,2)),(OCTLRM,OCMRT(1,6,2)),
     1(OCSLRM,OCTLRM(1,6,2)),(OCBOX,OCSLRM(1,6,2))
C
        INTEGER OCTLRM,OCSLRM,OCBOX,CBXPTR
        REAL NCRT,NCMRT
C
      DOUBLE PRECISION CNAME
C
      INTEGER TYPE,HSW,TSW,CSW,ASW,LSW,TGCNT,TRGNS,BXN,HZRGN,TRGT,BXTLRM
     1 ,BXSLRM,WSW,AACSW,TGBXN,DIGIT,DASH,COMMA,CE,CL,OUNIT
C
      REAL NTGRD,NTGRT
C
      DIMENSION TGBXN(25),OTGRD(25,10,2),OTGRT(25,10,2),NTGRD(25,10,2),
     1NTGRT(25,10,2)
C
        EQUIVALENCE (OTGRT,OTGRD(1,6,2)),(NTGRD,OTGRT(1,6,2)),
     1 (NTGRT,NTGRD(1,6,2))
C
        COMMON/BOX/TRD(10,2),ID(30,2),RSPDST(30,2),RTIME(10,2),
     1 CHANGE(10,2)
C
        INTEGER CHANGE
C
      DIMENSION CNAME(2),DIGIT(4),LNBUF(72)
        EQUIVALENCE (LNBUF,NTGRT(1,6,2)),
     1 (MXD,MAXDUE(1)),(MXD2,MAXDUE(2))
C
```

```
      DATA CNAME/6HENGINE,6HLADDER/,STAR/1H*/,BLANK/1H /
      DATA LNBUF(2)/1H /,DASH/1H-/,COMMA/1H,/,CE/1HE/,CL
     1 /1HL/, IRRDSW/1/,ZERO/0/
C
        WRITE(SYSOUT,1)
         BETAMX=BETAMX-15
C INITIALIZE UNITS
         IUNIT=DATA
       GO TO 110
C
100      REWIND IUNIT
C READ BOX NUMBERS,COORDINATES & CONTROL INFO FROM DATA BASE
110     READ(IUNIT) MAXPID,MAXDUE,NREG,NBOX,NEWID,RDSW,RDFCT,
     1 A,B,C,XDIST
        READ(IUNIT) (BOXNUM(I),X(I),Y(I),I=1,NBOX)
         TRGNS=NREG+2
120     WRITE(SYSOUT,12)
C GET USER'S COMMAND SEQUENCE
        CALL INPUT
C GET LAST COMMAND CODE; IC (INSTRUCTION COUNTER) ALWAYS POINTS TO LAST
C     COMMAND ON RETURN FROM 'INPUT',
         I=-BETACD(IC)
C EXECUTE COMMAND
       GO TO (900,200,900,900,300,600,800,700),I
C
C CANCEL COMMAND
C
C GET PARAMETER
200      I=BETACD(IC+1)
C 0 => CANCEL ALL
       IF (I .NE. 0) GO TO 210
         IC=1
       GO TO 120
C SET IC TO ITH TO LAST COMMAND
210      IC=IC-1
       IF(BETACD(IC) .GE. 0) GO TO 210
         I=I-1
       IF (I .GT. 0) GO TO 210
       GO TO 120
C
C OUTPUT COMMAND
C
C GET LIST SWITCH (1=NO, 2=YES)
300      LSW=BETACD(IC+8)
C MAP SWITCH (1=NO, 2=YES)
         MSW=BETACD(IC+10)
C WEIGHTING SWITCH (1=STRUCTURALS, 2=TOTAL ALARMS)
         WSW=BETACD(IC+9)
```

```
C AFFECTED REGION SWITCH (1=YES, 0=NO)
         ASW=BETACD(IC+7)
C COMPANY RESPONSE REGION SWITCH (1=YES, 0=NO)
         CSW=BETACD(IC+6)
C TARGET HAZARD SWITCH (1=YES, 0=NO)
         TSW=BETACD(IC+5)
C HAZARD(DEMAND) REGION SWITCH (1=YES, 0=NO)
         HSW=BETACD(IC+4)
C ALARM ASSIGNMENT OUTPUT SWITCH
         AACSW=BETACD(IC+11)
C OUTPUT UNIT
         OUNIT=BETACD(IC+12)
         IF(OUNIT .EQ. 0) GO TO 301
C INSURE OUTPUT FOR BOTH COMPANY TYPES
         BETACD(IC+1)=1
         BETACD(IC+2)=1
C INSURE VALID UNIT ASSIGNMENT
         IF(OUNIT .NE. DATA .AND. OUNIT .NE. IUNIT) GO TO 301
         WRITE(SYSOUT,72) IUNIT,OUNIT
         GO TO 120
C DUENESS
301      IDUE=BETACD(IC+3)
C       BETACD(IC+1) IS ENGINE OUTPUT SWITCH
C       BETACD(IC+2) IS LADDER OUTPUT SWITCH
C       (1=YES, 0=NO)
C INITIALIZE NUMBER OF LISTS ALLOCATED
         ILIST=0
         IF(MSW .NE. 2) GO TO 302
         ILIST=2
302      IF(AACSW .NE. 2) GO TO 3022
         WRITE(AAC,30)
C INITIALIZE NUMBER OF TARGET HAZARDS
3022     TGCNT=0
C INITIALIZE STAT ACCUMULATORS AND LIST POINTERS
          DO 310 K=1,2
C SKIP INITIALIZATION IF NO OUTPUT REQUESTED FOR COMPANY TYPE K
       IF (BETACD(IC+K) .EQ. 0) GO TO 310
C INSURE ALARM ASSIGNMENT INFO EXISTS FOR REQUESTED DUENESS
       IF (IDUE .LE. MAXDUE(K)) GO TO 3025
         SFX=ORDSFX(MAXDUE(K))
         WRITE (SYSOUT,17) IMX,SFX,CNAME(K),IDUE
       GO TO 100
C INITIALIZE ONLY UP TO DUENESS REQUESTED
3025     ISTAT=IDUE
         IF(OUNIT .NE. 0) ISTAT=MAXDUE(K)
         DO 308 J=1,ISTAT
C INITIALIZE TRAVEL TIME HISTOGRAMS
       DO 303 I=1,31
```

```
       DO 303 L=1,7
303      RHIST(I,L,J,K)=0
C INITIALIZE COMPANY STATS
       DO 304 I=1,MAXPID
         CBXPTR(I,J,K)=0
         NCRT(I,J,K)=0.0
         NCMRT(I,J,K)=0.0
         NCTLRM(I,J,K)=0
         NCSLRM(I,J,K)=0
         NCBOX(I,J,K)=0
304    CONTINUE
C INITIALIZE DEMAND REGION STATS
       DO 305 I=1,TRGNS
         NRRD(I,J,K)=0.0
         NRWRD(I,J,K)=0.0
         NRRT(I,J,K)=0.0
         NRWRT(I,J,K)=0.0
         NRMRT(I,J,K)=0.0
         NRSWRD(I,J,K)=0.
         NRSWRT(I,J,K)=0.
305    CONTINUE
C ASSIGN BOX LISTS FOR COMPANY AND AFFECTED REGIONS AS REQUESTED
        IF( LSW .LT. 2) GO TO 3065
       DO 306 I=1,2
       IF (BETACD(IC+5+I) .EQ. 0) GO TO 306
         ILIST=ILIST+1
         IF(ILIST .LE. NLISTS) GO TO 3055
         WRITE(SYSOUT,15)
         GO TO 120
C FOR 'LIST', I=1 FOR COMPANY RESPONSE AREAS, 2 FOR AFFECTED BOXES
3055     LIST(I,J,K)=ILIST
306     CONTINUE
3065   IF (ASW .EQ. 0) GO TO 308
C INITIALIZE AFFECTED REGION STATS
        N=(NREG+1)*2
       DO 307 I=1,N
         AFNBOX(I,J,K)=0
         AFTLRM(I,J,K)=0
         AFSLRM(I,J,K)=0
        ABXPTR(I,J,K)=0
        I1=TRGNS+I
         ORRD(I1,J,K)=0.0
         ORWRD(I1,J,K)=0.0
         ORRT(I1,J,K)=0.0
         ORWRT(I1,J,K)=0.0
         ORMRT(I1,J,K)=0.0
        NRRD(I1,J,K)=0.
        NRWRD(I1,J,K)=0.
```

```
          NRRT(I1,J,K)=0.
          NRWRT(I1,J,K)=0.
          NRMRT(I1,J,K)=0.
307     CONTINUE
308     CONTINUE
310     CONTINUE
C
C PROCESS BOX DATA
C
        DO 390 IBX=1,NBOX
C READ DEMAND REGION, TARGET HAZARD INDICATOR, INCIDENT ALARM ASSIGNMENT
C      AND RESPONSE DISTANCE DATA.
          READ (IUNIT) HZRGN,TRGT,BXTLRM,BXSLRM,((ID(J,K),RSPDST(J,K),
     1 J=1,MXD),K=1,2)
C SAVE BOX NUMBER IF TARGET HAZARD AND TARGET HAZARD OUTPUT REQUESTED
        IF(TSW .EQ. 0 .OR. TRGT .EQ. 0)GO TO 315
          TGCNT=TGCNT+1
          TGBXN(TGCNT)=IBX
C SAVE CURRENT TRAVEL DATA
315     DO 320 K=1,2
          TNEWID(K)=NEWID
        IF (BETACD(IC+K) +MSW .LT. 2) GO TO 320
        DO 317 J=1,IDUE
          RD=RSPDST(J,K)
         CHANGE(J,K)=0
        IF (TSW .EQ. 0 .OR. TRGT .EQ. 0) GO TO 317
          OTGRD(TGCNT,J,K)=RD
          OTGRT(TGCNT,J,K)=RSPTM(RD)
317       TRD(J,K)=RD
         N=MAXDUE(K)+1
         DO 318 J=N,MAXPID
         ID(J,K)=MAXPID
318      RSPDST(J,K)=1E11
320     CONTINUE
C
C APPLY SAVED CHANGES TO BOX
C
         CALL UPDATE(IBX,IERR)
         IF(IERR .NE. 0) GO TO 100
C
C POST STATISTICS
C
370     CONTINUE
          IF(OUNIT .EQ. 0 .OR. IBX .NE. 1) GO TO 371
C WRITE INFO FOR NEW DATA BASE IF REQUESTED
          NID=MAX0(NEWID,TNEWID(1),TNEWID(2))
          REWIND OUNIT
        WRITE(OUNIT) MAXPID,MAXDUE,NREG,
```

```
     1 NBOX,NID,RDSW,RDFCT,A,B,C,XDIST
        WRITE(OUNIT) (BOXNUM(I),X(I),Y(I),I=1,NBOX)
C GET ALARM COUNT ACCORDING TO WEIGHTING REQUESTED
371      BXALRM=BXSLRM
         IF (WSW .EQ. 2) BXALRM=BXTLRM
C ACCUMULATE STATS FOR COMPANY TYPES REQUESTED
       DO 386 K=1,2
       IF (BETACD(IC+K) +MSW .LT. 2 .AND. OUNIT .EQ. 0) GO TO 386
C ACCUMULATE STATS UP TO DUENESS REQUESTED
         ISTAT=IDUE
         IF(OUNIT .NE. 0)ISTAT=MAXDUE(K)
       DO 384 J=1,ISTAT
C GET WEIGHTED & UNWEIGHTED RESPONSE DISTANCE AND TRAVEL TIME
         RD=RSPDST(J,K)
         AWRD=RD*BXTLRM
         SWRD=RD*BXSLRM
         RT=RSPTM(RD)
         AWRT=RT*BXTLRM
         SWRT=RT*BXSLRM
C SAVE TRAVEL TIME FOR MAPPING DATA
         RTIME(J,K)=RT
C DETERMINE TRAVEL TIME HISTOGRAM SUBSCRIPT
        IB=31
        IF( RT .LE. 15.) IB=INT(2.*RT)+1
C POST CITYWIDE HISTOGRAM
         RHIST(IB,1,J,K)=RHIST(IB,1,J,K)+1
C POST PROPOSED CITYWIDE STATS
          CALL RPOST(2,J,K,WSW)
       IF (TRGT .NE. 1 .OR. (TSW .NE. 1 .AND. OUNIT .EQ. 0)) GO TO 376
C  TARGET HAZARD STATS
          CALL RPOST(1,J,K,WSW)
         NTGRD(TGCNT,J,K)=RD
         NTGRT(TGCNT,J,K)=RT
C AFFECTED REGION STATS
376      ORD=TRD(J,K)
        CHANGE(J,K)=0
       IF (RD .EQ. ORD) GO TO 378
C I=1 FOR DEGRADED RESPONSE, I=2 FOR IMPROVED
C
         I=2
         IF (RD .GT. ORD)I=1
        CHANGE(J,K)=I*2-3
      IF (ASW .EQ. 0) GO TO 378
         ORT=RSPTM(ORD)
        N=HZRGN
C POST CURRENT AND PROPOSED AFFECTED REGION STATS
       DO 377 II=1,2
         IF (II .EQ. 2) N=0
```

```
        L=I+2*N
         AFNBOX(L,J,K)=AFNBOX(L,J,K)+1
         AFTLRM(L,J,K)=AFTLRM(L,J,K)+BXTLRM
         AFSLRM(L,J,K)=AFSLRM(L,J,K)+BXSLRM
        L=L+TRGNS
         ORRD(L,J,K)=ORRD(L,J,K)+ORD
         ORWRD(L,J,K)=ORWRD(L,J,K)+ORD*BXALRM
         ORRT(L,J,K)=ORRT(L,J,K)+ORT
         ORWRT(L,J,K)=ORWRT(L,J,K)+ORT*BXALRM
         IF (ORT .GT. ORMRT(L,J,K)) ORMRT(L,J,K)=ORT
377       CALL RPOST(L,J,K,WSW)
       IF (LSW .EQ. 1) GO TO 378
C ADD BOX TO APPROPRIATE AFFECTED LIST
         ILIST=LIST(2,J,K)
         BXLST(IBX,ILIST)=ABXPTR((HZRGN-1)*2+I,J,K)
         ABXPTR((HZRGN-1)*2+I,J,K)=IBX
C DEMAND REGION STATS
378    CONTINUE
       IF(HSW .EQ. 0 .AND. OUNIT .EQ. 0) GO TO 380
         I=HZRGN+1
          CALL RPOST(HZRGN+2,J,K,WSW)
         RHIST(IB,I,J,K)=RHIST(IB,I,J,K)+1
C COMPANY STATS
380    IF (CSW .EQ. 0 .AND. OUNIT .EQ. 0) GO TO 384
         IID=ID(J,K)
         NCRT(IID,J,K)=NCRT(IID,J,K)+RT
         IF (NCMRT(IID,J,K) .LT. RT) NCMRT(IID,J,K)=RT
         NCTLRM(IID,J,K)=NCTLRM(IID,J,K)+BXTLRM
         NCSLRM(IID,J,K)=NCSLRM(IID,J,K)+BXSLRM
         NCBOX(IID,J,K)=NCBOX(IID,J,K)+1
       IF (LSW .EQ. 1) GO TO 384
         BXLST(IBX,LIST(1,J,K))=CBXPTR(IID,J,K)
         CBXPTR(IID,J,K)=IBX
384    CONTINUE
386    CONTINUE
C WRITE MAPPING DATA IF REQUIRED
         IF(MSW .EQ. 2) WRITE(MAP,38) BOXNUM(IBX),X(IBX),Y(IBX),HZRGN,
     1 BXTLRM,BXSLRM,((RTIME(J,K),CHANGE(J,K),K=1,2),J=1,IDUE)
C WRITE BOX RECORD FOR NEW DATA BASE IF REQUESTED
         IF(OUNIT .NE. 0) WRITE(OUNIT) HZRGN,TRGT,BXTLRM,BXSLRM,
     1 ((ID(J,K),RSPDST(J,K),J=1,MXD),K=1,2)
         IF(AACSW .NE. 2) GO TO 390
         WRITE(AAC,39) BOXNUM(IBX),(ID(I,1),I=1,MXD)
         WRITE(AAC,40) (ID(I,2),I=1,MXD2)
390    CONTINUE
C
C OUTPUT SECTION
C
```

```
400     CONTINUE
C READ SUMMARY STATS IF FIRST TIME AROUND
        IF (IRRDSW .EQ. 0) GO TO 402
          IRRDSW=0
         READ(IUNIT)
     1  (((OCRT(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1  (((OCMRT(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1  (((OCTLRM(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1  (((OCSLRM(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1  (((OCBOX(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2)
         READ(IUNIT)
     1  (((ORRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORWRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORWRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORMRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORSWRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (((ORSWRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1  (RGNBOX(I),RGTLRM(I),RGSLRM(I),I=1,TRGNS)
C
C PRINT OUTPUT FOR COMPANY TYPES REQUESTED
402     DO 490 K=1,2
        IF (BETACD(IC+K) .NE. 1) GO TO 490
C CITYWIDE AND DEMAND REGION OUTPUT
        DO 407 J=1,IDUE
          SFX=ORDSFX(J)
         WRITE(SYSOUT,4) J ,SFX,CNAME(K)
C PRINT CITYWIDE STATS
           CALL RPRINT(2,J,K,WSW)
        IF (HSW .EQ. 0) GO TO 406
C PRINT DEMAND REGION STATS
        DO 405 I=3,TRGNS
           CALL RPRINT(I,J,K,WSW)
405     CONTINUE
C PRINT CITYWIDE HISTOGRAM
406        CALL DPRINT(1,RHIST(1,1,J,K))
        IF (HSW .EQ. 0) GO TO 407
          II=NREG+1
C PRINT DEMAND REGION HISTOGRAMS
        DO 4061 I=2,II
        IF(RGNBOX(I+1) .EQ. 0) GO TO 4061
           CALL DPRINT(I,RHIST(1,I,J,K))
4061    CONTINUE
407     CONTINUE
410     IF (CSW .EQ. 0) GO TO 420
C PRINT COMPANY STATS
        DO 417 J=1,IDUE
          SFX=ORDSFX(J)
```

```
          WRITE (SYSOUT,4) J,SFX,CNAME(K)
         CALL CPRINT(J,K,LSW)
417      CONTINUE
420     CONTINUE
        IF (ASW .EQ. 0) GO TO 430
C AFFECTED REGION OUTPUT
        DO 425 J=1,IDUE
          SFX=ORDSFX(J)
          WRITE (SYSOUT,4) J,SFX,CNAME(K)
        IF (AFNBOX(1,J,K) + AFNBOX(2,J,K) .NE. 0) GO TO 423
         WRITE(SYSOUT,42)
        GO TO 425
C PRINT OUTPUT FOR ALL 3 AFFECTED REGIONS AFTER COMPUTING
C       CURRENT STATS.
423      CALL APRINT(J,K,WSW,LSW)
425     CONTINUE
430     CONTINUE
        IF (TSW .EQ.0 .OR. TGCNT .EQ.0) GO TO 490
C TARGET HAZARD OUTPUT
        DO 435 J=1,IDUE
          SFX=ORDSFX(J)
         WRITE(SYSOUT,4) J,SFX,CNAME(K)
         CALL RPRINT (1,J,K,WSW)
         WRITE(SYSOUT,10)
        DO 435 I=1,TGCNT
          CIND=BLANK
          IF (OTGRT(I,J,K) .NE. NTGRT(I,J,K))CIND=STAR
          WRITE (SYSOUT,11) CIND,BOXNUM(TGBXN(I)),OTGRD(I,J,K),
     1 NTGRD(I,J,K),OTGRT(I,J,K),NTGRT(I,J,K)
435     CONTINUE
490     CONTINUE
C
C COMPUTE & WRITE SUMMARY STATS FOR NEW DATA BASE IF REQUESTED
C
          IF(OUNIT .EQ. 0) GO TO 100
          DO 540 K=1,2
          IDUE=MAXDUE(K)
          DO 535 J=1,IDUE
          DO 530 I=1,MAXPID
          BOX=NCBOX(I,J,K)
          IF(BOX .EQ. 0.) GO TO 530
          NCRT(I,J,K)=NCRT(I,J,K)/BOX
530       CONTINUE
535       CONTINUE
540       CONTINUE
         WRITE(OUNIT)
     1 (((NCRT(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1 (((NCMRT(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
```

```
     1 (((NCTLRM(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1 (((NCSLRM(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2),
     1 (((NCBOX(I,J,K),I=1,MAXPID),J=1,MXD),K=1,2)
        WRITE(OUNIT)
     1 (((NRRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1 (((NRWRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1 (((NRRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1 (((NRWRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1 (((NRMRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1 (((NRSWRD(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1 (((NRSWRT(I,J,K),I=1,TRGNS),J=1,MXD),K=1,2),
     1 (RGNBOX(I),RGTLRM(I),RGSLRM(I),I=1,TRGNS)
C
         REWIND OUNIT
       GO TO 100
C
C PRINT COMMAND
C
600      IIC=1
         ICNT=0
       DO 605 K=1,2
605      TNEWID(K)=NEWID
C
610    IF (IIC .EQ. IC) GO TO 120
C GET NEXT COMMAND CODE
         I=-BETACD(IIC)
         ICNT=ICNT+1
         IIC=IIC+1
         K=BETACD(IIC)
       GO TO (620,999,640,660),I
C ADD
620     WRITE(SYSOUT,62) ICNT,CNAME(K),TNEWID(K),BETACD(IIC+1)
         TNEWID(K)=TNEWID(K)+1
625      IIC=IIC+4
         K=BETACD(IIC)
       IF (K .LT. 0) GO TO 610
        WRITE(SYSOUT,63) CNAME(K),TNEWID(K),BETACD(IIC+1)
         TNEWID(K)=TNEWID(K)+1
       GO TO 625
C DELETE
640     WRITE(SYSOUT,64) ICNT,CNAME(K),BETACD(IIC+1)
645      IIC=IIC+2
         K=BETACD(IIC)
       IF (K .LT. 0) GO TO 610
        WRITE(SYSOUT,65) CNAME(K),BETACD(IIC+1)
       GO TO 645
C MOVE
660     WRITE(SYSOUT,66) ICNT,CNAME(K),BETACD(IIC+1),BETACD(IIC+2)
```

```
665        IIC=IIC+5
           K=BETACD(IIC)
        IF (K .LT. 0) GO TO 610
         WRITE(SYSOUT,67) CNAME(K),BETACD(IIC+1),BETACD(IIC+2)
        GO TO 665
C
C INPUT COMMAND
C
700        IRRDSW=1
           IUNIT=BETACD(IC+1)
           IF(IUNIT .EQ. 0) IUNIT=DATA
           GO TO 100
C
C EXIT COMMAND
C
800        STOP
900       WRITE(SYSOUT,999)
           STOP
1       FORMAT(25X,'NEW YORK CITY - RAND INSTITUTE'/24X,
       1'FIRE HOUSE SITE EVALUATION MODEL')
12      FORMAT(1H0)
15      FORMAT(' ***INSUFFICIENT STORAGE FOR BOX LISTS'/
       1' REDUCE OUTPUT OPTIONS OR RECOMPILE')
17      FORMAT(' *** ONLY ',I2,A2,' DUE ',A6,'S ASSIGNED;'/
       1'       CANNOT CONTINUE WITH DUE DEPTH ',I2)
38         FORMAT(I4,2F8.5,I2,2I4,200(F5.2,I3))
4       FORMAT(1H1,I2,A2,' DUE ',A6,' RESPONSE')
42      FORMAT(//' NO AFFECTED BOXES')
10      FORMAT(//1H0,3X,'BOX    TRV. DIST.   TRV. TIME'/10X,2('CURR. PROP.
       1  '))
11      FORMAT(1H0,A1,2X,I4,2(1X,2(1X,F5.2)))
62      FORMAT(1H0,I2,')  ADD ',A6,1X,I2,' AT BOX ',I4)
63      FORMAT(6X,'ADD ',A6,1X,I2,' AT BOX ',I4)
64      FORMAT(1H0,I2,')  DELETE ',A6,1X,I2)
65      FORMAT (6X,'DELETE ',A6,1X,I2)
66      FORMAT (1H0,I2,')  MOVE ',A6,1X,I2,' TO BOX ',I4)
67      FORMAT (6X,'MOVE ',A6,1X,I2,' TO BOX ',I4)
999     FORMAT('0***INTERNAL ERROR;  INVALID COMMAND CODE FROM INPUT.')
30         FORMAT(' BOX')
39         FORMAT(1H0,I4,' ENGINES'/(5X,19(1X,I3)))
40         FORMAT(6X,'LADDERS'/(5X,19(1X,I3)))
72         FORMAT(' *** I = ',I2,', O = ',I2,' NOT VALID ASSIGN',
       1'MENT - REENTER.')
        END
```

```
BLOCK.FOR[4016,10]        13:23    23-MAY-75        PAGE 1

C
      BLOCK DATA
C
        COMMON/LISTS/NLISTS,LIST(2,10,2),BOXNUM(750),BXLST(750,10)
C
        DIMENSION X(750),Y(750)
C
        EQUIVALENCE (X,BXLST(1,1)),(Y,BXLST(1,2))
C
        INTEGER BOXNUM,BXLST
         COMMON/CODES/WORD,NUM,EQ,LP,RP,STR,END,CMNDS(7),TEMPL(4,2)
         COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
        INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
        COMMON/CV/IC,BETAMX,BETACD(200)
C
        INTEGER BETAMX,BETACD
C
      INTEGER WORD,EQ,RP,STR,END,CMNDS,TEMPL
C
      DATA WORD/1/,NUM/2/,EQ/3/,LP/4/,RP/5/,STR/6/,END/7/,CMNDS/1HA,1HC,
     11HD,1HM,1HO,1HP,1HE/,TEMPL/0,0,1,1,1,0,0,1/
      DATA DATA/31/,SYSOUT/5/,SYSIN/4/,MAP/32/,AAC/33/
      DATA IC/1/
      DATA NLISTS/10/,BETAMX/200/
         END
```

```
LOOKUP.FOR[4016,10]      13:23   23-MAY-75       PAGE 1

C
          FUNCTION LOOKUP(LIST,N,ARG)
C BINARY SEARCH ROUTINE
C
C INPUT:'LIST' - ARRAY TO BE SEARCHED: INTEGER VALUES IN ASCENDING ORDER
C         'N' - NUMBER OF ELEMENTS IN 'LIST'
C         'ARG' - SEARCH ARGUEMENT
C
C RETURNS: SUBSCRIPT OF ELEMENT OF 'LIST' EQUAL TO 'ARG' OR 0 IF FAILED
C
      INTEGER ARG,HIGH
      DIMENSION LIST(N)
         LOW=1
         HIGH=N
       IF (ARG.GE.LIST(1) .AND.ARG .LT.LIST(N)) GO TO 100
         LOOKUP=0
        IF (ARG .LT. LIST(1) .OR. ARG .GT. LIST(N))RETURN
         LOOKUP=N
         RETURN
100      I=(HIGH+LOW)/2
       IF (ARG .GT. LIST(I)) GO TO 200
       IF (ARG.LT.LIST(I)) GO TO 300
         LOOKUP=I
         RETURN
200      IF (LOW .EQ. I) RETURN
         LOW =I
       GO TO 100
300      HIGH=I
       GO TO 100
         END
```

```
C COMMAND READER/INTERPRETER ROUTINE
C
C
C ACCEPTS USER'S COMMANDS AND SAVES THEM IN ENCODED FORM IN ARRAY BETACD
C RETURNS TO CALLING PROGRAM FOR COMMAND EXECUTION WHEN CANCEL, EXIT,
C    OUTPUT OR PRINT COMMAND IS ENCOUNTERED.
C
C ON ENTRY, 'IC' CONTAINS SUBSCRIPT OF FIRST AVAILABLE ELEMENT OF BETACD
C ON EXIT, 'IC' IS SUBSCRIPT OF LAST COMMAND ENTERED IN BETACD.
C
        SUBROUTINE INPUT
         COMMON/CODES/WORD,NUM,EQ,LP,RP,STR,END,CMNDS(7),TEMPL(4,2)
C
         COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
        INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
        COMMON/CITY/MAXPID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,C,XDIST,
     1 TNEWID(2),RDSW
        INTEGER TNEWID,RDSW
C
        COMMON/CV/IC,BETAMX,BETACD(200)
C
        INTEGER BETAMX,BETACD
C
        COMMON/LISTS/NLISTS,LIST(2,10,2),BOXNUM(750),BXLST(750,10)
C
        DIMENSION X(750),Y(750)
C
        EQUIVALENCE (X,BXLST(1,1)),(Y,BXLST(1,2))
C
        INTEGER BOXNUM,BXLST
C
      INTEGER WORD,EQ,RP,STR,END,CMNDS,TEMPL
C
      INTEGER CHARO,CHARC,CHARP,CHARE,CHARM,CHARA,CHARD,CHARU,CHARI,
     C CHARL,CHARR,CHARW,CHARH,CHART,CHARS,CHARN,CHARY
      DATA CHARO/1HO/,CHARC/1HC/,CHARP/1HP/,CHARE/1HE/,CHARM/1HM/,
     C CHARA/1HA/,CHARD/1HD/,CHARL/1HL/,CHARY/1HY/,CHARR/1HR/,
     C CHARW/1HW/,CHARH/1HH/,CHART/1HT/,CHARS/1HS/,CHARN/1HN/,
     C CHARU/1HU/,CHARI/1HI/
C
      INTEGER TYPE,VALUE,X,Y
C
C GET IDENTIFIER OF USER'S NEXT COMMAND
100       TYPE=END
          CALL TOKEN(TYPE,VALUE)
C INSURE CHARACTER VALUE
        IF (TYPE .EQ. WORD) GO TO 110
```

```
          WRITE (SYSOUT,1)
        GO TO 100
C SET TEMP IC
110       IIC=IC
C TEST FOR IMMEDIATE TYPE COMMAND
        IF (VALUE .EQ. CHARO) GO TO 500
        IF (VALUE .EQ. CHARC) GO TO 600
        IF (VALUE .EQ. CHARP) GO TO 700
        IF (VALUE .EQ. CHARE) GO TO 800
          IF(VALUE .EQ. CHARI) GO TO 900
C INSURE ENOUGH STORAGE AVAILABLE FOR CHANGE TYPE COMMAND
        IF (IIC .GT. BETAMX) GO TO 999
C TEST FOR CHANGE TYPE COMMAND
        IF(VALUE .EQ. CHARM) GO TO 200
        IF (VALUE .EQ. CHARA) GO TO 300
        IF (VALUE .EQ. CHARD) GO TO 400
          WRITE (SYSOUT,2)
        GO TO 100
C
C SET 'M'OVE CODE
C
200       I=4
C BRANCH TO COMMON CHANGE TYPE COMMAND ROUTINE
        GO TO 405
C
C SET 'A'DD CODE
C
300       I=1
        GO TO 405
C
C SET 'D'ELETE CODE
400       I=3
C CHANGE TYPE COMMAND COMMON ROUTINE
C
C SAVE COMMAND CODE; MINUS INDICATES NEW COMMAND
405       BETACD(IC)=-I
C POINT TO FIRST PARAMETER STORAGE ELEMENT
          IIC=IC+1
C GET COMPANY TYPE
           CALL TOKEN(TYPE,VALUE)
C INSURE SUFFICIENT ROOM FOR COMMAND
410     IF (IIC .GT. BETAMX) GO TO 999
        IF (TYPE .EQ. WORD) GO TO 415
          WRITE (SYSOUT,3)
        GO TO 100
415     IF (VALUE .NE. CHARE) GO TO 420
          BETACD(IIC)=1
        GO TO 430
```

```
420     IF(VALUE .EQ. CHARL) GO TO 425
         WRITE(SYSOUT,3)
        GO TO 100
425       BETACD(IIC)=2
430       IIC=IIC+1
C SEE IF COMPANY ID REQUIRED
        IF (TEMPL(I,1) .NE. 1) GO TO 445
C GET COMPANY ID
           CALL TOKEN(TYPE,VALUE)
        IF (TYPE .EQ. NUM) GO TO 435
         WRITE(SYSOUT,43)
        GO TO 100
435     IF (VALUE .LT. NEWID) GO TO 440
          WRITE (SYSOUT,4301) VALUE
        GO TO 100
440       BETACD(IIC)=VALUE
          IIC=IIC+1
C SEE IF BOX NUM NEEDED
445     IF (TEMPL(I,2) .NE. 1) GO TO 460
           CALL TOKEN(TYPE,VALUE)
        IF (TYPE .EQ. NUM) GO TO 450
         WRITE(SYSOUT,44)
        GO TO 100
C FIND INTERNAL BOX NUMBER
450       IBX=LOOKUP(BOXNUM,NBOX,VALUE)
        IF (IBX .NE. 0) GO TO 455
          WRITE (SYSOUT,45) VALUE
        GO TO 100
C SET EXTERNAL BOX NUM IN STACK
455       BETACD(IIC)=VALUE
          IIC=IIC+1
C SET BOX COORDINATES IN STACK
          BETACD(IIC)=X(IBX)
          IIC=IIC+1
          BETACD(IIC)=Y(IBX)
          IIC=IIC+1
C GET NEXT COMPANY TYPE OR END OF STATEMENT
460        CALL TOKEN(TYPE,VALUE)
        IF(TYPE .NE. END) GO TO 410
C UPDATE IC TO INDICATE COMMAND IS COMPLETE
          IC=IIC
        GO TO 100
C
C 'O'UTPUT COMMAND
C
500     CONTINUE
          BETACD(IC)=-5
C SET DEFAULT OPTIONS
```

```
        DO 515 I=1,11
515       BETACD(IC+I)=1
          BETACD(IC+12)=0
C GET PARAMETER KEYWORD
520         CALL TOKEN(TYPE,VALUE)
         IF (TYPE .EQ. END) RETURN
        IF (TYPE .EQ. WORD) GO TO 525
         WRITE(SYSOUT,8)
        GO TO 100
C BRANCH TO PARAMETER DRIVER
525     IF (VALUE .EQ. CHARD) GO TO 530
        IF (VALUE .EQ. CHARR) GO TO 550
        IF (VALUE .EQ. CHARL) GO TO 580
        IF (VALUE .EQ. CHARC) GO TO 540
        IF (VALUE .EQ. CHARW) GO TO 570
        IF (VALUE .EQ. CHARM) GO TO 579
          IF(VALUE .EQ. CHARA) GO TO 578
          IF(VALUE .EQ. CHARU) GO TO 593
          WRITE (SYSOUT,8)
        GO TO 100
C 'D'UE PARM; GET DUE DEPTH
530         CALL TOKEN(TYPE,VALUE)
C INSURE VALID DUE DEPTH
         IF (TYPE .EQ. NUM .AND. VALUE .LE. MAXDUE(1) .AND. VALUE .GT. 0)
     1GO TO 535
         WRITE(SYSOUT,9)
        GO TO 100
535       BETACD(IC+3)=VALUE
        GO TO 520
C 'C'OMPANY PARM
C CHECK FOR LEFT PAREN
540         CALL TOKEN(TYPE,VALUE)
        IF (TYPE .EQ. LP) GO TO 542
          WRITE (SYSOUT,3)
        GO TO 100
C TURN OFF BOTH COMPANY TYPE FLAGS
542       BETACD(IC+1)=0
          BETACD(IC+2)=0
C GET NEXT ELEMENT
544         CALL TOKEN(TYPE,VALUE)
C DONE IF RIGHT PAREN
        IF (TYPE .EQ. RP) GO TO 520
C INTERPRET COMPANY TYPE
        IF (TYPE .EQ. WORD) GO TO 546
         WRITE(SYSOUT,3)
        GO TO 100
546       I=0
          IF (VALUE .EQ. CHARE)I=1
```

```
          IF (VALUE .EQ. CHARL) I=2
         IF (I .NE. 0) GO TO 548
          WRITE(SYSOUT,9)
         GO TO 100
C TURN ON FLAG FOR SELECTED COMPANY TYPE
548       BETACD(IC+I)=1
         GO TO 544
C 'R'EGION PARM;
550        CALL TOKEN(TYPE,VALUE)
C INSURE FIRST ELEMENT LPAREN
         IF (TYPE .EQ. LP) GO TO 552
          WRITE(SYSOUT,13)
         GO TO 100
C TURN OFF REGION FLAGS
552      DO 554 I=4,7
554       BETACD(IC+I)=0
C GET NEXT 'R' PARM ELEMENT
556        CALL TOKEN(TYPE,VALUE)
C DONE IF RPAREN
         IF (TYPE .EQ. RP) GO TO 520
         IF (TYPE .EQ. WORD) GO TO 558
          WRITE(SYSOUT,14)
         GO TO 100
558       I=0
C DETERMINE REGION SELECTED
          IF (VALUE .EQ. CHARD) I=4
          IF (VALUE .EQ. CHART) I=5
          IF (VALUE .EQ. CHARC) I=6
          IF (VALUE .EQ. CHARA)I=7
         IF (I .NE. 0) GO TO 560
          WRITE (SYSOUT,14)
         GO TO 100
C SET FLAG FOR REGION SELECTED
560       BETACD(IC+I)=1
         GO TO 556
C 'W'EIGHTING PARM
C GET PARM VALUE
570        CALL TOKEN(TYPE,VALUE)
         IF (TYPE .EQ. WORD) GO TO 572
          WRITE(SYSOUT,13)
         GO TO 100
C STRUCTURAL WEIGHTING IS DEFAULT; RETURN IF PARM VALUE IS 'S'
572      IF (VALUE .EQ. CHARS) GO TO 520
         IF (VALUE .EQ. CHARA) GO TO 574
          WRITE(SYSOUT,13)
         GO TO 100
C INDICATE TOTAL ALARM WEIGHTING
574       BETACD(IC+9)=2
```

```
        GO TO 520
C 'A'AC PARM
578       I=11
          GO TO 581
C 'M'AP PARM
579       I=10
        GO TO 581
C 'L'IST PARM
580       I=8
C GET PARM VALUE
581        CALL TOKEN(TYPE,VALUE)
        IF (TYPE .EQ. WORD) GO TO 585
         WRITE(SYSOUT,10)
        GO TO 100
C NO LISTS AND NO MAP IS DEFAULT
585     IF (VALUE .EQ. CHARN) GO TO 520
        IF (VALUE .EQ. CHARY) GO TO 590
         WRITE(SYSOUT,10)
        GO TO 100
C INDICATE MAP OR LIST OPTION SELECTED
590       BETACD(IC+I)=2
        GO TO 520
C 'U'NIT PARM
593       CALL TOKEN(TYPE,VALUE)
          IF(TYPE .NE. NUM) GO TO 925
          IF(VALUE .GT. 99) GO TO 925
          BETACD(IC+12)=VALUE
          GO TO 520
C
C 'C'ANCEL COMMAND
C
600     CONTINUE
          IERR=0
          BETACD(IC)=-2
C GET CANCEL PARM
           CALL TOKEN(TYPE,VALUE)
C SET 0 FOR 'CANCEL ALL'
          BETACD(IC+1)=0
C EXIT IF PARM IS '*' FOR 'ALL'
         IF (TYPE .EQ. STR) RETURN
        IF (TYPE .EQ. NUM) GO TO 610
         WRITE(SYSOUT,11)
        GO TO 100
C SET NUMBER OF CHANGES TO DELETE
610       BETACD(IC+1)=VALUE
          RETURN
C
C 'P'RINT COMMAND
```

```
C
700       BETACD(IC)=-6
          RETURN
C
C 'E'XIT COMMAND
C
800       BETACD(IC)=-7
          RETURN
C
C 'I'NPUT COMMAND
C
900       BETACD(IC)=-8
          CALL TOKEN(TYPE,VALUE)
          IF (TYPE .EQ. WORD) GO TO 910
905       WRITE(SYSOUT,91)
91        FORMAT(' *** INVALID INPUT PARM - REENTER.')
          GO TO 100
910       IF(VALUE .NE. CHARU) GO TO 905
          CALL TOKEN(TYPE,VALUE)
          IF(TYPE .EQ. NUM) GO TO 930
925       WRITE(SYSOUT,92)
92        FORMAT(' ***INVALID UNIT NO. - REENTER.')
          GO TO 100
930       IF(VALUE .GT. 99) GO TO 925
          BETACD(IC+1)=VALUE
         RETURN
C
C
999      WRITE(SYSOUT,99)
        GO TO 100
1      FORMAT(' *** COMMAND NOT ALPHABETIC - REENTER.')
2      FORMAT(' *** INVALID COMMAND - REENTER.')
3      FORMAT(' *** INVALID COMPANY TYPE, MUST BE ''E'' OR ''L'' - REENTE
      1R.')
43     FORMAT(' *** COMPANY ID MUST BE NUMERIC -  REENTER.')
4301   FORMAT(' *** COMPANY ID ',I3,' OUT OF RANGE - REENTER.')
44     FORMAT(' *** INVALID BOX NUMBER - REENTER.')
45     FORMAT(' *** BOX ',I5,' NOT IN DATA BASE - REENTER.')
8      FORMAT(' *** INVALID PARAMETER SPECIFIED - REENTER.')
9      FORMAT(' *** INVALID DUE DEPTH - REENTER.')
14     FORMAT(' *** INVALID REGION SPEC. - REENTER.')
13     FORMAT(' ***INVALID WEIGHT OPTION, MUST BE ''S'' OR ''A'' - REENTE
      1R.')
10     FORMAT(' ***INVALID LIST OR MAP OPTION; MUST BE Y OR N - REENTER.'
      1)
11     FORMAT(' ***INVALID CLEAR PARAMETER - REENTER.')
99     FORMAT(' *** NO SPACE AVAILABLE TO SAVE CHANGES - ONLY O, C, P OR
      1E COMMANDS ACCEPTED.')
```

```
      END
```

```
TOKEN,FOR[4016,10]        13:23    23-MAY-75        PAGE 1

C ROUTINE TO GET NEXT INPUT ELEMENT
C
C RETURNS 'TYPE' AND 'VALUE', 'TYPE' IS CLASS OF ELEMENT AND CAN ASSUME
C       VALUES:
C       1) NUM - NUMERIC 'VALUE' (INTEGER)
C       2) WORD - LETTER 'VALUE'
C       3) LP - LEFT PAREN (NO VALUE)
C       4) RP - RIGHT PAREN (NO 'VALUE')
C       5) STR - ASTERISK (NO 'VALUE')
C       6) END - END OF INPUT LINE (NO VALUE)
C
          SUBROUTINE TOKEN(TYPE,VALUE)
          COMMON/CODES/WORD,NUM,EQ,LP,RP,STR,END,CMNDS(7),TEMPL(4,2)
C
          COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
         INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
       INTEGER WORD,EQ,RP,STR,END,CMNDS,TEMPL
C
       DIMENSION CARD(81)
       INTEGER TYPE,VALUE,ALPHMN,ALPHMX,COL,CARD,RDUNIT,CHAR,RPAREN,STAR
       DATA ALPHMN/1HA/,ALPHMX/1HZ/,NUMIN/1H0/,NUMAX/1H9/,CARD(81)
      1/1H /,LPAREN/1H(/,RPAREN/1H)/,EQUAL/1H=/,STAR/1H*/
C
         DIMENSION IDIGIT(10)
         DATA IDIGIT/1H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/
C READ NEW LINE IF LAST LINE FINISHED
        IF (TYPE .NE. END .AND. COL .LT. 81) GO TO 200
          WRITE (SYSOUT,2)
         READ(SYSIN,1) (CARD(I),I=1,80)
          COL=1
200       TYPE=END
C EXAMINE NEXT COLUMN
210       CHAR=CARD(COL)
C CHECK FOR LETTER ELEMENT
        IF (CHAR .GE. ALPHMN .AND. CHAR .LE.ALPHMX) GO TO 300
C CHECK FOR NUMERIC ELEMENT
        IF (CHAR .GE. NUMIN .AND. CHAR .LE. NUMAX) GO TO 400
          IF (CHAR .EQ. LPAREN) TYPE=LP
          IF (CHAR .EQ.RPAREN) TYPE=RP
          IF (CHAR .EQ. STAR) TYPE=STR
          COL=COL+1
C LOOK AT NEXT COLUMN IF NO ELEMENT FOUND
        IF (COL .LT. 81 .AND. TYPE .EQ. END) GO TO 210
          RETURN
C LETTER ELEMENT:
300       TYPE=WORD
          VALUE=CHAR
```

```
TOKEN.FOR[4016,10]       13:23   23-MAY-75        PAGE 2

         COL=COL+1
         RETURN
C NUMERIC ELEMENT:
400      VALUE=0
C CONVERT DIGIT STRING TO BINARY INTEGER
410     DO 415 I=1,10
        IF(CHAR .EQ. IDIGIT(I)) GO TO 420
415     CONTINUE
        TYPE=NUM
        RETURN
420     VALUE=VALUE*10+(I-1)
        COL=COL+1
        CHAR=CARD(COL)
        GO TO 410
2     FORMAT(' COMMAND? '$)
1     FORMAT(80A1)
      END
```

```
ORDSFX.FOR[4016,10]      13:23   23-MAY-75        PAGE 1

C
C ROUTINE TO DETERMINE ORDINAL SUFFIX OF INTEGER 'I' TO BE PRINTED IN A2
C    FORMAT.
          FUNCTION ORDSFX(I)
      DIMENSION SUFFIX(4)
      DATA SUFFIX/2HST,2HND,2HRD,2HTH/
          J=MOD(I,10)
       IF(I/10 .NE. 1 .AND. J .LT. 4 .AND. J .GT. 0) GO TO 100
          ORDSFX=SUFFIX(4)
          RETURN
100       ORDSFX=SUFFIX(J)
          RETURN
          END
```

RPOST.FOR[4016,10]   13:23   23-MAY-75   PAGE 1

```
C SUBROUTINE TO POST 'J'TH DUE COMPANY TYPE 'K' STATISTICS FOR
C REGION 'I'. 'WSW' IS TYPE OF WEIGHTING(1=STRUCTURAL, 2=TOTAL).
        SUBROUTINE RPOST(I,J,K,WSW)
C
        COMMON/CITY/MAXPID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,C,XDIST,
     1 TNEWID(2),RDSW
        INTEGER TNEWID,RDSW
C
        COMMON/NEWREG/RD,RT,AWRD,AWRT,SWRD,SWRT,RHIST(31,7,10,2)
C
        DIMENSION NRRD(22,10,2),NRWRD(22,10,2),NRRT(22,10,2),
     1 NRWRT(22,10,2),NRMRT(22,10,2),NRSWRD(8,10,2),NRSWRT(8,10,2)
C
        EQUIVALENCE (NRRD,RHIST(1,1,6,2)),(NRWRD,NRRD(1,6,2)),
     1 (NRRT,NRWRD(1,6,2)),(NRWRT,NRRT(1,6,2)),(NRMRT,NRWRT(1,6,2)),
     1 (NRSWRD,NRMRT(1,6,2)),(NRSWRT,NRSWRD(1,6,2))
C
        REAL NRRD,NRWRD,NRRT,NRWRT,NRMRT,NRSWRT,NRSWRD
C
        INTEGER RHIST
C
        INTEGER WSW
C
        NRRD(I,J,K)=NRRD(I,J,K)+RD
        NRRT(I,J,K)=NRRT(I,J,K)+RT
        IF(RT .GT. NRMRT(I,J,K)) NRMRT(I,J,K)=RT
C FOR AFFECTED REGIONS, DON'T NEED TWO TYPES OF WEIGHTING
        IF(I .GT. NREG+2) GO TO 100
        NRWRD(I,J,K)=NRWRD(I,J,K)+AWRD
        NRWRT(I,J,K)=NRWRT(I,J,K)+AWRT
        NRSWRT(I,J,K)=NRSWRT(I,J,K)+SWRT
        NRSWRD(I,J,K)=NRSWRD(I,J,K)+SWRD
        RETURN
100     IF(WSW .EQ. 1) GO TO 110
        WD=AWRD
        WT=AWRT
        GO TO 120
110     WD=SWRD
        WT=SWRT
120     NRWRD(I,J,K)=NRWRD(I,J,K)+WD
        NRWRT(I,J,K)=NRWRT(I,J,K)+WT
        RETURN
        END
```

```
C
        SUBROUTINE RPRINT(I,J,K,WSW)
C ROUTINE TO PRINT 'J'TH DUE COMPANY TYPE 'K' STATISTICS FOR REGION 'I'
C WITH WEIGHTING ACCORDING TO 'WSW'
C
       COMMON/CITY/MAXPID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,C,XDIST,
     1 TNEWID(2),RDSW
       INTEGER TNEWID,RDSW
C
        COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
       INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
       COMMON/OLDREG/RGNBOX(8),RGTLRM(8),RGSLRM(8),AFNBOX(14,10,2)
C
       DIMENSION ORWRD(22,10,2),ORRT(22,10,2),ORWRT(22,10,2),
     1 ORMRT(22,10,2),ORSWRD(8,10,2),ORSWRT(8,10,2),
     1 ORRD(22,10,2),AFTLRM(14,10,2),AFSLRM(14,10,2),
     1 ABXPTR(14,10,2)
C
       EQUIVALENCE (AFTLRM,AFNBOX(1,6,2)),(AFSLRM,AFTLRM(1,6,2)),
     1 (ABXPTR,AFSLRM(1,6,2)),(ORRD,ABXPTR(1,6,2)),
     1 (ORWRD,ORRD(1,6,2)),(ORRT,ORWRD(1,6,2)),
     1 (ORWRT,ORRT(1,6,2)),(ORMRT,ORWRT(1,6,2)),
     1 (ORSWRD,ORMRT(1,6,2)),(ORSWRT,ORSWRD(1,6,2))
C
       INTEGER RGNBOX,RGTLRM,RGSLRM,AFNBOX,AFTLRM,AFSLRM
C
       COMMON/NEWREG/RD,RT,AWRD,AWRT,SWRD,SWRT,RHIST(31,7,10,2)
C
       DIMENSION NRRD(22,10,2),NRWRD(22,10,2),NRRT(22,10,2),
     1 NRWRT(22,10,2),NRMRT(22,10,2),NRSWRD(8,10,2),NRSWRT(8,10,2)
C
       EQUIVALENCE (NRRD,RHIST(1,1,6,2)),(NRWRD,NRRD(1,6,2)),
     1 (NRRT,NRWRD(1,6,2)),(NRWRT,NRRT(1,6,2)),(NRMRT,NRWRT(1,6,2)),
     1 (NRSWRD,NRMRT(1,6,2)),(NRSWRT,NRSWRD(1,6,2))
C
       REAL NRRD,NRWRD,NRRT,NRWRT,NRMRT,NRSWRT,NRSWRD
C
       INTEGER RHIST
C
      REAL NEW,NALRM,NBOX,NWRD,NWRT
      INTEGER WSW
      DOUBLE PRECISION AFNAME,WEIGHT
      DIMENSION AFNAME(3),WEIGHT(2,2)
      DATA AFNAME/8HDEGRADED,8HIMPROVED,8HAFFECTED/,WEIGHT/8HSTRUCTUR,6H
     1ALARMS,6HALS    ,6H
C
       INTEGER TRGNS
```

```
C
        TRGNS=NREG+2
        II=I
        IF(I .GE. NREG+3) GO TO 500
        IF(I .EQ. 2) GO TO 200
        IF(I .LT. 0) GO TO 400
        IF(I .GT. 2) GO TO 300
C
C TARGET HAZARD REGION
C
        NBOX=RGNBOX(1)
        IF(NBOX .LT. 1.) RETURN
        WRITE(SYSOUT,3) RGNBOX(1),RGTLRM(1),RGSLRM(1)
        GO TO 350
C
C CITYWIDE REGION
C
200     NBOX=RGNBOX(2)
        IF(NBOX .LT. 1.) RETURN
        WRITE(SYSOUT,2) RGNBOX(2),RGTLRM(2),RGSLRM(2)
        GO TO 350
C
C DEMAND REGION I-2
C
300     NBOX=RGNBOX(I)
        IF(NBOX .LT. 1.) RETURN
        IM2=I-2
        WRITE(SYSOUT,1) IM2,RGNBOX(I),RGTLRM(I),RGSLRM(I)
350     IABS=I
        IF(WSW .EQ. 1) GO TO 360
        NALRM=RGTLRM(I)
        OWRT=ORWRT(I,J,K)
        NWRT=NRSWRT(I,J,K)
        OWRD=ORSWRD(I,J,K)
        NWRD=NRSWRD(I,J,K)
        GO TO 600
360     NALRM=RGSLRM(I)
        OWRT=ORSWRT(I,J,K)
        NWRT=NRSWRT(I,J,K)
        OWRD=ORSWRD(I,J,K)
        NWRD=NRSWRD(I,J,K)
        GO TO 600
C
C OVERALL AFFECTED BOXES IN REGION I
C
400     IABS=-I
        NBOX=AFNBOX(IABS-TRGNS,J,K)
        IF(NBOX .LT. 1.) RETURN
```

```
        INAME=3
        GO TO 510
C
C DEGRADED OR IMPROVED BOXES IN REGION I
C
500     NBOX=AFNBOX(I-TRGNS,J,K)
        IF(NBOX .LT. 1.) RETURN
        IABS=I
        INAME=MOD(IABS-NREG-3,2)+1
510     WRITE(SYSOUT,4) NBOX,AFNAME(INAME),AFTLRM(IABS-TRGNS,J,K),AFSLRM
     1 (IABS-TRGNS,J,K)
        IF(WSW .EQ. 1) GO TO 520
        NALRM=AFTLRM(IABS-TRGNS,J,K)
        GO TO 530
520     NALRM=AFSLRM(IABS-TRGNS,J,K)
530     OWRT=ORWRT(IABS,J,K)
        OWRD=ORWRD(IABS,J,K)
        NWRT=NRWRT(IABS,J,K)
        NWRD=NRWRD(IABS,J,K)
C
C COMMON PRINT ROUTINE
C
600     OLD=ORRT(IABS,J,K)/NBOX
        NEW=NRRT(IABS,J,K)/NBOX
        WRITE(SYSOUT,5) OLD,NEW
        OLD=ORRD(IABS,J,K)/NBOX
        NEW=NRRD(IABS,J,K)/NBOX
        WRITE(SYSOUT,6) OLD,NEW
        IF(NALRM .EQ. 0.) GO TO 605
        OWRT=OWRT/NALRM
        NWRT=NWRT/NALRM
                OWRD=OWRD/NALRM
        NWRD=NWRD/NALRM
605     WRITE(SYSOUT,7) (WEIGHT(WSW,L),L=1,2),OWRT,NWRT
        WRITE(SYSOUT,8) (WEIGHT(WSW,L),L=1,2),OWRD,NWRD
        WRITE(SYSOUT,9) ORMRT(IABS,J,K),NRMRT(IABS,J,K)
C
         RETURN
1      FORMAT('OREGION ',I2,2X,'(',I4,' BOXES) CURRENT  PROPOSED (',I5,'
     1ALARMS, ',I4,' STRUCTS.)')
2      FORMAT('OCITYWIDE  (',I4,' BOXES)  CURRENT  PROPOSED (',I5,' ALARM
     1S, ',I4,' STRUCTS.)')
3      FORMAT('0',I2,' TARGET HAZARDS',7X,'CURRENT  PROPOSED (',I5,' ALAR
     1MS, ',I4,' STRUCTS.)')
4      FORMAT('0',I4,1X,A8,' BOXES',5X,'CURRENT  PROPOSED (',I5,' ALARMS,
     1 ',I4,' STRUCTS.)')
5      FORMAT(' AV. TR.T.',12X,2(4X,F5.2))
6      FORMAT(' AV. TR.D.',12X,2(4X,F5.2))
```

```
7       FORMAT(' AV. TR.T. TO ',A8,A3,1X,F5.2,4X,F5.2)
8       FORMAT(' AV. TR.D. TO ',A8,A3,1X,F5.2,4X,F5.2)
9       FORMAT(' MAX TR.T.',12X,2(4X,F5.2))
        END
```

```
BXPRNT.FOR[4016,10]      13:23    23-MAY-75        PAGE 1

C ROUTINE TO PRINT A LINKED LIST OF ALARM BOX NUMBERS.
C
C 'LIST' IS A 1 DIMENSIONAL ARRAY EACH ELEMENT OF WHICH CORRESPONDS TO
C      AN ALARM BOX.
C  THE CORRESPONDING ELEMENT OF 'BOXNUM' CONTAINS THE EXTERNAL BOX
C      NUMBER.
C THE VALUE OF EACH ELEMENT IN 'LIST' IS THE SUBSCRIPT OF THE NEXT
C      ELEMENT TO BE PRINTED. A 0 VALUE FLAGS THE LAST ELEMENT.
C   'IPNT' CONTAINS THE SUBSCRIPT OF THE FIRST ELEMENT TO BE PRINTED.
C
          SUBROUTINE BXPRNT(ILIST,IPNT)
C
          COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
         INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
         COMMON/LISTS/NLISTS,LIST(2,10,2),BOXNUM(750),BXLST(750,10)
C
         DIMENSION X(750),Y(750)
C
         EQUIVALENCE (X,BXLST(1,1)),(Y,BXLST(1,2))
C
         INTEGER BOXNUM,BXLST
C
      DIMENSION ILIST(1),BXLINE(14)
      INTEGER BXLINE
         I=IPNT
         J=0
110    IF (I .EQ. 0) GO TO 130
         J=J+1
         BXLINE(J)=BOXNUM(I)
       IF(J .LT. 14) GO TO 120
        WRITE(SYSOUT,1) BXLINE
         J=0
120      I=ILIST(I)
       GO TO 110
130      IF (J .EQ. 0) RETURN
        WRITE(SYSOUT,1) (BXLINE(K),K=1,J)
         RETURN
1     FORMAT(14(1X,I4))
      END
```

```
RSPTM.FOR[4016,10]        13:23    23-MAY-75        PAGE 1

C TRAVEL TIME SPLINE FUNCTION.
C    'RD' IS RESPONSE DISTANCE
C
          FUNCTION RSPTM(RD)
C
         COMMON/CITY/MAXPID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,C,XDIST,
         1 TNEWID(2),RDSW
         INTEGER TNEWID,RDSW
C
        IF(RD .GT. XDIST) GO TO 100
          RSPTM=C*SQRT(RD)
          RETURN
100       RSPTM=A+B*RD
          RETURN
          END
```

```
DPRINT.FOR[4016,10]      13:23   23-MAY-75        PAGE 1

C ROUTINE TO PRINT TRAVEL TIME HISTOGRAM FOR REGION I
C
          SUBROUTINE DPRINT(I,HIST)
C
          COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
         INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
       DIMENSION HIST(31),HTTL(32)
       DATA HTTL/4H 0.0,4H 0.5,4H 1.0,4H 1.5,4H 2.0,4H 2.5,4H 3.0,4H 3.5,
      14H 4.0,4H 4.5,4H 5.0,4H 5.5,4H 6.0,4H 6.5,4H 7.0,4H 7.5,4H 8.0,4H
      18.5,
      14H 9.0,4H 9.5,4H10.0,4H10.5,4H11.0,4H11.5,4H12.0,4H12.5,4H13.0,4H1
      13.5,4H14.0,4H14.5,4H15.0,4H****/
       INTEGER HIST,HMIN,HMAX
C
        IF (I .GT. 1) GO TO 110
         WRITE(SYSOUT,1)
        GO TO 120
110       II=I-1
         WRITE(SYSOUT,2) II
C FIND LAST HISTOGRAM BUCKET TO PRINT
120       HMAX=32
125       HMAX=HMAX-1
        IF (HIST(HMAX) .EQ. 0) GO TO 125
C FIND FIRST BUCKET TO PRINT
          HMIN=0
130       HMIN=HMIN+1
        IF(HIST(HMIN) .EQ. 0) GO TO 130
C PRINT REQUIRED BUCKETS, 12 TO A LINE
140       II=HMIN+11
          IF (II .GT. HMAX) II=HMAX
          J=II+1
          WRITE (SYSOUT,3) (HTTL(K),K=HMIN,J)
          WRITE (SYSOUT,4) (HIST(K),K=HMIN,II)
          HMIN=J
        IF(HMIN .LE. HMAX) GO TO 140
          RETURN
1      FORMAT(///' DISTRIBUTION OF TRAVEL TIMES TO BOXES - CITYWIDE')
2      FORMAT(///' DISTRIBUTION OF TRAVEL TIMES TO BOXES - REGION ',I2)
3      FORMAT('OTR TIME',13(1X,A4))
4      FORMAT(' # BOXES  ',12(2X,I3))
       END
```

```
CPRINT.FOR[4016,10]      13:23   23-MAY-75        PAGE 1

C
C SUBROUTINE TO PRINT COMPANY RESPONSE AREA STATISTICS FOR
C 'J'TH DUE COMPANY TYPE 'K' STATISTICS. 'LSW' INDICATES
C WHETHER OR NOT LISTS OF BOXES IN THE COMPANY RESPONSE
C AREAS WILL BE PRINTED.
C
        SUBROUTINE CPRINT(J,K,LSW)
C
         COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
        INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
        COMMON/LISTS/NLISTS,LIST(2,10,2),BOXNUM(750),BXLST(750,10)
C
        DIMENSION X(750),Y(750)
C
        EQUIVALENCE (X,BXLST(1,1)),(Y,BXLST(1,2))
C
        INTEGER BOXNUM,BXLST
C
        COMMON/COMP/NCRT(30,10,2)
C
      DIMENSION OCRT(30,10,2),OCMRT(30,10,2),OCTLRM(30,10,2),
     1OCSLRM(30,10,2),OCBOX(30,10,2),
     1NCMRT(30,10,2),NCTLRM(30,10,2),NCSLRM(30,10,2),NCBOX(30,10,2),
     1CBXPTR(30,10,2)
C
      EQUIVALENCE (NCMRT,NCRT(1,6,2)),(NCTLRM,NCMRT(1,6,2)),
     1(NCSLRM,NCTLRM(1,6,2)),(NCBOX,NCSLRM(1,6,2)),(CBXPTR,NCBOX(1,6,2))
     1,(OCRT,CBXPTR(1,6,2)),(OCMRT,OCRT(1,6,2)),(OCTLRM,OCMRT(1,6,2)),
     1(OCSLRM,OCTLRM(1,6,2)),(OCBOX,OCSLRM(1,6,2))
C
        INTEGER OCTLRM,OCSLRM,OCBOX,CBXPTR
        REAL NCRT,NCMRT
C
        COMMON/CITY/MAXPID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,C,XDIST,
     1 TNEWID(2),RDSW
        INTEGER TNEWID,RDSW
C
        DOUBLE PRECISION CNAME
        DIMENSION CNAME(2)
        DATA BLANK/1H /,STAR/1H*/,ZERO/0./,
     1 CNAME(1)/6HENGINE/,CNAME(2)/6HLADDER/
C
         WRITE (SYSOUT,6)
         II=NEWID-1
C PRINT STATS FOR CURRENT COMPANIES
       DO 413 I=1,II
         BX=NCBOX(I,J,K)
```

```
        IF (OCBOX(I,J,K) .EQ. 0 .AND. BX .EQ. 0.) GO TO 413
          RT=0.
          IF (BX .NE. 0.) RT=NCRT(I,J,K)/BX
          CIND=BLANK
        IF(OCBOX(I,J,K) .NE. NCBOX(I,J,K) .OR. RT .NE. OCRT(I,J,K))
     1CIND=STAR
          WRITE (SYSOUT,7) CIND,I,OCBOX(I,J,K),NCBOX(I,J,K),OCRT(I,J,K),
     1RT,OCMRT(I,J,K),NCMRT(I,J,K),OCTLRM(I,J,K),NCTLRM(I,J,K),
     1OCSLRM(I,J,K),NCSLRM(I,J,K)
413     CONTINUE
C PRINT STATS FOR ANY ADDED COMPANIES
        DO 414 I=NEWID,MAXPID
          BX=NCBOX(I,J,K)
        IF(BX .EQ. 0) GO TO 414
          RT=NCRT(I,J,K)/BX
         WRITE(SYSOUT,7) STAR,I,ZERO,NCBOX(I,J,K),ZERO,RT,ZERO,NCMRT(I,J,
     1K),ZERO,
     1 NCTLRM(I,J,K),ZERO,NCSLRM(I,J,K)
414     CONTINUE
        IF (LSW .EQ.1) RETURN
C PRINT COMPANY BOX LISTS
        DO 415 I=1,MAXPID
        IF (NCBOX(I,J,K) .EQ. 0) GO TO 415
          WRITE (SYSOUT,9) CNAME(K),I
           CALL BXPRNT(BXLST(1,LIST(1,J,K)),CBXPTR(I,J,K))
415     CONTINUE
         RETURN
7       FORMAT(1H0,A1,2X,I2,1X,2(2X,I3),1X,2(1X,2(1X,F5.2)),2(1X,2(2X,I4))
     1)
9       FORMAT(1H0,A6,' COMPANY ',I2,' BOXES:')
6       FORMAT(1H0,3X,'CO.     BOXES      AV. TR.T.     MAX TR.T.     ALARMS
     1      STRUCTURALS',
     1/6X,5('  CURR. PROP.'))
          END
```

```
APRINT.FOR[4016,10]      13:23    23-MAY-75        PAGE 1

C
C SUBROUTINE TO PRINT 'J'TH DUE COMPANY TYPE 'K' AFFECTED REGION
C STATISTICS. 'WSW' SPECIFIES THE TYPE OF WEIGHTING TO BE USED
C FOR WEIGHTED AVERAGES AND 'LSW' INDICATES WHETHER OR NOT LISTS
C OF THE AFFECTED BOXES IN EACH DEMAND REGION ARE TO BE PRINTED.
C
        SUBROUTINE APRINT(J,K,WSW,LSW)
C
        COMMON/OLDREG/RGNBOX(8),RGTLRM(8),RGSLRM(8),AFNBOX(14,10,2)
C
        DIMENSION ORWRD(22,10,2),ORRT(22,10,2),ORWRT(22,10,2),
     1 ORMRT(22,10,2),ORSWRD(8,10,2),ORSWRT(8,10,2),
     1 ORRD(22,10,2),AFTLRM(14,10,2),AFSLRM(14,10,2),
     1 ABXPTR(14,10,2)
C
        EQUIVALENCE (AFTLRM,AFNBOX(1,6,2)),(AFSLRM,AFTLRM(1,6,2)),
     1 (ABXPTR,AFSLRM(1,6,2)),(ORRD,ABXPTR(1,6,2)),
     1 (ORWRD,ORRD(1,6,2)),(ORRT,ORWRD(1,6,2)),
     1 (ORWRT,ORRT(1,6,2)),(ORMRT,ORWRT(1,6,2)),
     1 (ORSWRD,ORMRT(1,6,2)),(ORSWRT,ORSWRD(1,6,2))
C
        INTEGER RGNBOX,RGTLRM,RGSLRM,AFNBOX,AFTLRM,AFSLRM,ABXPTR
C
        COMMON/NEWREG/RD,RT,AWRD,AWRT,SWRD,SWRT,RHIST(31,7,10,2)
C
        DIMENSION NRRD(22,10,2),NRWRD(22,10,2),NRRT(22,10,2),
     1 NRWRT(22,10,2),NRMRT(22,10,2),NRSWRD(8,10,2),NRSWRT(8,10,2)
C
        EQUIVALENCE (NRRD,RHIST(1,1,6,2)),(NRWRD,NRRD(1,6,2)),
     1 (NRRT,NRWRD(1,6,2)),(NRWRT,NRRT(1,6,2)),(NRMRT,NRWRT(1,6,2)),
     1 (NRSWRD,NRMRT(1,6,2)),(NRSWRT,NRSWRD(1,6,2))
C
        REAL NRRD,NRWRD,NRRT,NRWRT,NRMRT,NRSWRT,NRSWRD
C
        INTEGER RHIST
C
         COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
        INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
        COMMON/LISTS/NLISTS,LIST(2,10,2),BOXNUM(750),BXLST(750,10)
C
        DIMENSION X(750),Y(750)
C
        EQUIVALENCE (X,BXLST(1,1)),(Y,BXLST(1,2))
C
        INTEGER BOXNUM,BXLST
C
        COMMON/CITY/MAXPID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,C,XDIST,
```

```
     1 TNEWID(2),RDSW
        INTEGER TNEWID,RDSW
C
        IR=-1
        N1=NREG+3
        N2=NREG*3+3
C INDEX THROUGH CITYWIDE AND ALL DEMAND REGIONS, LOOKING AT
C AFFECTED BOX STATISTICS.
        DO 100 IREG=N1,N2,2
        IR=IR+1
        IR2=IR*2
        IR2P1=IR2+1
        IR2P2=IR2+2
        IF(IR .GT. 0) GO TO 20
        WRITE(SYSOUT,9001)
9001    FORMAT('0CITYWIDE')
        GO TO 30
20      IF(AFNBOX(IR2P1,J,K)+AFNBOX(IR2P2,J,K) .LT. 1) GO TO 100
        WRITE(SYSOUT,9002) IR
9002    FORMAT('0REGION ',I2)
30      CONTINUE
        ISUPP=0
C INDEX THROUGH DEGRADED AND IMPROVED BOX STATISTICS
        DO 60 I=1,2
        IF(AFNBOX(IR2+I,J,K) .NE. 0) GO TO 40
        ISUPP=1
        GO TO 60
40      CALL RPRINT(IREG+I-1,J,K,WSW)
60      CONTINUE
        IF(ISUPP .NE. 0) GO TO 70
C ADD IMPROVED BOX STATISTICS TO DEGRADED BOX STATISTICS TO GET
C OVERALL AFFECTED BOX STATISTICS FOR THE REGION.
        IP1=IREG+1
        AFNBOX(IR2P1,J,K)=AFNBOX(IR2P1,J,K)+AFNBOX(IR2P2,J,K)
        AFTLRM(IR2P1,J,K)=AFTLRM(IR2P1,J,K)+AFTLRM(IR2P2,J,K)
        AFSLRM(IR2P1,J,K)=AFSLRM(IR2P1,J,K)+AFSLRM(IR2P2,J,K)
        ORRD(IREG,J,K)=ORRD(IREG,J,K)+ORRD(IP1,J,K)
        ORWRD(IREG,J,K)=ORWRD(IREG,J,K)+ORWRD(IP1,J,K)
        ORRT(IREG,J,K)=ORRT(IREG,J,K)+ORRT(IP1,J,K)
        ORWRT(IREG,J,K)=ORWRT(IREG,J,K)+ORWRT(IP1,J,K)
        ORMRT(IREG,J,K)=AMAX1(ORMRT(IREG,J,K),ORMRT(IP1,J,K))
        NRRD(IREG,J,K)=NRRD(IREG,J,K)+NRRD(IP1,J,K)
        NRWRD(IREG,J,K)=NRWRD(IREG,J,K)+NRWRD(IP1,J,K)
        NRRT(IREG,J,K)=NRRT(IREG,J,K)+NRRT(IP1,J,K)
        NRWRT(IREG,J,K)=NRWRT(IREG,J,K)+NRWRT(IP1,J,K)
        NRMRT(IREG,J,K)=AMAX1(NRMRT(IREG,J,K),NRMRT(IP1,J,K))
C NEGATIVE REGION NUMBER INDICATES OVERALL AFFECTED TO RPRINT
        CALL RPRINT(-IREG,J,K,WSW)
```

```
70      IF(LSW .EQ. 1 .OR. IR .LT. 1) GO TO 100
C PRINT AFFECTED BOX LISTS FOR THE REGION IF REQUESTED
        I=ABXPTR(IR2-1,J,K)
        IF(I .NE. 0) WRITE(SYSOUT,9003)
        CALL BXPRNT(BXLST(1,LIST(2,J,K)),I)
        I=ABXPTR(IR2,J,K)
        IF(I .NE. 0) WRITE(SYSOUT,9004)
        CALL BXPRNT(BXLST(1,LIST(2,J,K)),I)
100     CONTINUE
        RETURN
9003    FORMAT('0DEGRADED BOXES:')
9004    FORMAT('0IMPROVED BOXES:')
        END
```

```
C
C SUBROUTINE TO UPDATE ALARM ASSIGNMENT LISTS WITH USER-SPECIFIED
C CONFIGURATION CHANGES.
C
C 'IBX' IS THE INTERNAL (RELATIVE) BOX NUMBER TO WHICH CHANGES ARE TO
C BE APPLIED. 'IERR' IS A FLAG WHICH INDICATES WHETHER OR NOT AN
C ERROR HAS BEEN ENCOUNTERED DURING THE UPDATING PROCESS.
C
        SUBROUTINE UPDATE(IBX,IERR)
C
        COMMON/LISTS/NLISTS,LIST(2,10,2),BOXNUM(750),BXLST(750,10)
C
        DIMENSION X(750),Y(750)
C
        EQUIVALENCE (X,BXLST(1,1)),(Y,BXLST(1,2))
C
        INTEGER BOXNUM,BXLST
C
         COMMON/SYSTEM/DATA,SYSOUT,SYSIN,MAP,AAC
        INTEGER DATA,SYSOUT,SYSIN,MAP,AAC
C
        COMMON/CV/IC,BETAMX,BETACD(200)
C
        INTEGER BETAMX,BETACD
C
        COMMON/CITY/MAXPID,MAXDUE(2),NREG,NBOX,NEWID,RDFCT,A,B,C,XDIST,
     1 TNEWID(2),RDSW
        INTEGER TNEWID,RDSW
C
        COMMON/BOX/TRD(10,2),ID(30,2),RSPDST(30,2),RTIME(10,2),
     1 CHANGE(10,2)
C
        INTEGER CHANGE
C
        INTEGER ADD,DEL,XNEW,YNEW
        DATA ADD/-1/,DEL/-3/
        EQUIVALENCE (TX,XNEW),(TY,YNEW)
        IERR=0
C SET TEMP INSTRUCTION COUNTER
         IIC=1
C IF ALL CHANGES PROCESSED, RETURN
320     IF (IIC .EQ. IC) RETURN
C GET COMMAND CODE
         I=-BETACD(IIC)
C SAVE POINTER TO CURRENT COMMAND
         LIC=IIC
C BRANCH TO COMMAND PROCESSOR
       GO TO (350,900,330,330),I
```

```
C
C DELETE AND MOVE
C
C POINT TO NEXT COMMAND ELEMENT
330       IIC=IIC+1
C TEST FOR NEXT COMMAND
        IF (BETACD(IIC) .LT. 0) GO TO 320
C GET COMPANY TYPE
          K=BETACD(IIC)
          IIC=IIC+1
C SKIP PROCESSING IF NO OUTPUT FOR COMPANY TYPE
        IF(BETACD(IC+K) .NE. 1) GO TO 333
C GET COMPANY ID
          IID=BETACD(IIC)
          IMX=MAXDUE(K)
C SEE IF COMPANY ASSIGNED TO BOX
        DO 332 J=1,MAXPID
        IF (IID .EQ. ID(J,K)) GO TO 334
332     CONTINUE
C COMPANY NOT ASSIGNED; CHECK DISTANCES IF MOVE, OR GET NEXT ELEMENT IF
     1DELETE.
333     IF (BETACD(LIC) .EQ. DEL ) GO TO 330
        GO TO 354
C DELETE COMPANY FROM ALARM ASSIGNMENT
334       I=MAXPID-1
        IF(I .LT. J) GO TO 338
        DO 336 JJ=J,I
          ID(JJ,K)=ID(JJ+1,K)
336       RSPDST(JJ,K)=RSPDST(JJ+1,K)
338       ID(MAXPID,K)=MAXPID
          RSPDST(IMX,K)=1E11
        IF (BETACD(LIC) .EQ. DEL) GO TO 330
        GO TO 354
C
C ADD
C
350       IIC=IIC+1
        IF (BETACD(IIC) .LT. 0) GO TO 320
C GET COMPANY TYPE
          K=BETACD(IIC)
        IF(TNEWID(K) .LE. MAXPID-1) GO TO 352
          WRITE (SYSOUT,3)
         IERR=1
         RETURN
C GET ID OF ADDED COMAPNY
352       IID=TNEWID(K)
C INCREMENT ADDED COMPANY ID FOR NEXT ADD
          TNEWID(K)=TNEWID(K)+1
```

UPDATE,FOR[4016,10]   13:23   23-MAY-75   PAGE 3

```
          IMX=MAXDUE(K)
C SKIP BOX NUMBER IN CODE
354       IIC=IIC+2
C GET COORDINATES OF BOX; NOTE TX AND TY EQUIVALENCED TO XNEW AND YNEW
C      TO PREVENT TYPE CONVERSION.
          XNEW=BETACD(IIC)
          IIC=IIC+1
        IF (BETACD(IC+K) .NE. 1) GO TO 357
          YNEW=BETACD(IIC)
C DETERMINE NEW RESPONSE DISTANCE FROM COMPANY TO BOX
        IF (RDSW .EQ. 2) GO TO 355
          DSTNEW=SQRT((TX-X(IBX))**2 + (TY-Y(IBX))**2)*RDFCT
        GO TO 3555
355       DSTNEW=ABS(TX-X(IBX))+ABS(TY-Y(IBX))
C SEE IF NEW RESPONSE DISTANCE LESS THAN ANY CURRENT RESPONSE DISTANCE
3555    DO 356 J=1,MAXPID
        IF (DSTNEW .LT. RSPDST(J,K)) GO TO 358
356     CONTINUE
357     IF(BETACD(LIC) .EQ. ADD) GO TO 350
        GO TO 330
C MAKE ROOM FOR NEW COMPANY
358       I=J+1
        IF (MAXPID .LE. J) GO TO 362
          ICT=MAXPID
        DO 360 JJ=I,IMX
          ICT1=ICT-1
          ID(ICT,K)=ID(ICT1,K)
          RSPDST(ICT,K)=RSPDST(ICT1,K)
360       ICT=ICT1
C INSERT NEW COMPNAY IN ALARM ASSIGNMENT
362       ID(J,K)=IID
          RSPDST(J,K)=DSTNEW
        IF(BETACD(LIC) .EQ. ADD) GO TO 350
        GO TO 330
900      WRITE(SYSOUT,4)
         STOP
4        FORMAT('0*** INTERNAL ERROR: INVALID CODE FROM INPUT - ',
     1 'EXECUTION TERMINATED')
3       FORMAT('0*** INSUFFICIENT STORAGE FOR ADDED CO.''S - RESPECIFY OR
     1RECOMPILE')
         END
```

Appendix F

CALCULATION OF OUTPUT MEASURES

We will use the following notation to explain the calculation of the output measures:

n_i The number of alarm boxes in region i.

d_{mijk} Travel distance from box m in region i to its jth-due company of type k (k = 1 for engines; k = 2 for ladders).

t_{mijk} Travel time from box m in region i to its jth-due company of type k (k = 1 for engines; k = 2 for ladders).

w_{mi} The total number of alarms or structural fires (user option) that occurred in the last year at box m in region i.

For each region i, response level j, and company type k for which the user has requested output, the program computes:

(1) $$(\text{average travel time})_{ijk} = \frac{\sum_{m=1}^{n_i} t_{mijk}}{n_i}$$

(2) $$(\text{average travel distance})_{ijk} = \frac{\sum_{m=1}^{n_i} d_{mijk}}{n_i}$$

(3) $$(\text{average weighted travel time})_{ijk} = \frac{\sum_{m=1}^{n_i} t_{mijk} w_{mi}}{\sum_{m=1}^{n_i} w_{mi}}$$

(4). $$(\text{average weighted travel distance})_{ijk} = \frac{\sum_{m=1}^{n_i} d_{mijk} w_{mi}}{\sum_{m=1}^{n_i} w_{mi}}$$

(5) $$(\text{maximum travel time})_{ijk} = \max\ (t_{1ijk}, t_{2ijk}, \ldots, t_{n_i ijk}).$$

Appendix G
CROSS-REFERENCED LISTING OF PROGRAM SEGMENTS

(ALL SYMBOLS)

SYMBOL	DEFINED IN	REFERENCED IN
APRINT	APRINT	MAIN.
BOX	MAIN.	UPDATE
BXPRNT	BXPRNT	CPRINT,APRINT
CITY	MAIN.	UPDATE,CPRINT,APRINT,RPOST,RPRINT,RSPTM,INPUT
CODES	MAIN.	DAT.,INPUT,TOKEN
COMP	MAIN.	CPRINT
CPRINT	CPRINT	MAIN.
CV	MAIN.	UPDATE,DAT.,INPUT
DAT.	DAT.	
DPRINT	DPRINT	MAIN.
INPUT	INPUT	MAIN.
LISTS	MAIN.	UPDATE,CPRINT,APRINT,BXPRNT,DAT.,INPUT
LOOKUP	LOOKUP	INPUT
MAIN.	MAIN.	
NEWREG	MAIN.	APRINT,RPOST,RPRINT
OLDREG	MAIN.	APRINT,RPRINT
ORDSFX	ORDSFX	MAIN.
RPOST	RPOST	MAIN.
RPRINT	RPRINT	MAIN.,APRINT
RSPTM	RSPTM	MAIN.
SYSTEM	MAIN.	UPDATE,CPRINT,APRINT,RPRINT,BXPRNT,DPRINT,DAT.,INPUT,TOKEN
TOKEN	TOKEN	INPUT
UPDATE	UPDATE	MAIN.

REFERENCES

1. Walker, W. and E. Ignall, "Methods for Analyzing the Deployment of Fire Companies," The New York City-Rand Institute, forthcoming.

2. Rider, K., "A Parametric Model for the Allocation of Fire Companies," The New York City-Rand Institute, R-1615-NYC/HUD, May 1975.

3. Carter, G., J. Chaiken, and E. Ignall, "Simulation Model of Fire Department Operations: Executive Summary," The New York City-Rand Institute, R-1188/1-HUD, December 1974.

4. Carter, G., "Simulation Model of Fire Department Operations: Program Description," The New York City-Rand Institute, R-1188/2-HUD, December 1974.

5. Hausner, J. and W. Walker, "An Analysis of the Deployment of Fire-Fighting Resources in Trenton, New Jersey," The New York City-Rand Institute, R-1566/1-TRNTN, February 1975.

6. Hausner, J., W. Walker, and A. Swersey, "An Analysis of the Deployment of Fire-Fighting Resources in Yonkers, New York," The New York City-Rand Institute, R-1566/2-HUD/CY, October 1974.

7. Rider, K. and J. Hausner, "An Analysis of the Deployment of Fire-Fighting Resources in Jersey City, New Jersey," The New York City-Rand Institute, R-1566/4-HUD, forthcoming.

8. Walker, W., D. Singleton, and B. Smith, "An Analysis of the Deployment of Fire-Fighting Resources in Wilmington, Delaware," The New York City-Rand Institute, R-1566/5-HUD, forthcoming.

9. Hausner, J., "Determining the Travel Characteristics of Emergency Service Vehicles," The New York City-Rand Institute, R-1687-HUD, April 1975.

10. Kolesar, P. and W. Walker, "Measuring the Response Characteristics of New York City's Fire Companies," The New York City-Rand Institute, R-1449-NYC, April 1974.

11. U. S. Bureau of the Census, "Census Use Study: Grids, A Computer Mapping System," U. S. Government Printing Office, Washington, D. C., 1972.