figure is proportional to the number of workers using the line to commute to work and selecting each station. The number of commuters is shown on the figure's briefcase. This display, while rather whimsical, may prove useful in communicating results to the public. This graphic is processed by using another property of the relative coordinate system. By multiplying all elements of a two-dimensional array of relative coordinates by a single factor, one can enlarge or reduce the object to be represented.

CONCLUSIONS

The software developed for this research takes advantage of the capabilities of interactive computer graphics in several ways. First, the quick turnaround of the system permits the analyst to explore a larger number of alternatives in a given time than is possible for a batch-mode computer model. The ease with which a model run can be performed encourages the user to explore a wide range of solutions and gives the user the opportunity to follow analytical paths that might not otherwise have been pursued. Subsequent runs can be made quickly, so that an idea can be tested while it is still fresh.

Second, the graphic form of the output shows at a glance the results of the model run. By using hard copies made from images on the terminal screen, one can immediately acquire an understanding of how results change by comparing visual outputs from one run to the next. One can pick out major shifts much more quickly from examining visual images than from examining numerical output. In addition, the pictorial quality of the output allows policymakers and non-technical individuals to grasp the implications of the analysis with greater clarity than can be achieved through examining reams of computer printout.

In an academic environment, the use of conversational computer graphics programs allowed undergraduates with little computer training to participate in the research. Undergraduates received 2 h of classroom instruction and then were assigned various policies to test. The easy manipulation of model parameters and sophisticated output gave students the sense of using a powerful lever and increased their motivation for working on the project.

ACKNOWLEDGMENT

This research was partially funded under a contract from the Program of University Research, U.S. Department of Transportation. We wish to thank Robert J. Ravera, Director, Office of University Research, and Richard I. Cohen, Office of Policy and Program Development, Urban Mass Transportation Administration, for their support and encouragement of this project.

Publication of this paper sponsored by Committee on Computer Graphics and Interactive Computing.

Network Simulation Interactive Computer Graphics Program

SHIH-MIAO CHIN AND AMIR EIGER

An overview of the deficiencies of the network simulation (NETSIM) program with regard to data input, data debugging, and analysis of the output is presented. Interactive computer graphic (ICG) enhancements are suggested as measures to eliminate many of the difficulties. The NETSIM/ICG program, which provides ICG capability in input-data preparation, input-data display, and both real-time and passive displays of link-specific measures of effectiveness, is described. The interactive data input, both graphical and keyboard in free format, follows a systematic procedure for obtaining the necessary information needed by NETSIM without reference to the user's manual. By using input-data display and input-data modification capabilities provided by the pre-NETSIM (PRENET) enhancement program, the user can easily comprehend and debug the NETSIM input data. As a consequence, significant reductions in the costs associated with data preprocessing are anticipated. The capabilities of providing both real-time and passive displays enable the user to more easily assimilate the information generated by the NETSIM simulation model and to comprehend the overall operation of the network. Consequently, these programs provide a heuristic approach to determining high-performance solutions at a minimal cost of both personnel and computer time.

As traffic flows through street networks, it experiences periods of congestion that may result from inadequate geometric design, signalization, or simply excessive demand. Traffic-simulation techniques are important tools for the traffic engineer in investigating the impacts of various traffic-control strategies. These simulation experiments can yield an enormous amount of data that could not be obtained in real life for economic or other reasons.

Among the network traffic simulation models, the network simulation (NETSIM) model produced for the Federal Highway Administration has been the most popular. NETSIM is an extension of the UTCS-1/SCOT simulation model, developed originally as an analytical tool for studying computer control of urban traffic networks. It has been extensively validated and is generally considered to yield reasonable results. The program is a microscopic simulation that deals with the movement of individual vehicles in an urban street network according to car-following, queue-discharge, and lane-changing theories. NETSIM defines the traffic network in terms of streets (links) and intersections (nodes). Each vehicle that travels through the network has associated with it data for, among other things, current speed, acceleration, and position. Detailed information concerning the operational characteristics of NETSIM may be found elsewhere (1,2).

THE PROBLEM

Although the NETSIM model is useful for accurately simulating traffic flows within an urban street network, certain deficiencies quickly become evident. These can be classified into three groups--data input, data debugging, and output analysis.

NETSIM is quite expensive to operate. Short (15-min) simulation runs of even a relatively small network generally require large amounts of computer time. Furthermore, input-data preparation requires excessive personnel time. One study (1) shows that 85 percent of the total cost of an initial NETSIM run consists of information-coding costs. For succeeding runs, about 65 percent of the total costs is in input-data modifications. There are several reasons for such high costs associated with input-data preparation. First, intuitive physical meaning of the network geometry and signal information is often lost, since all that information must be digitized. The coder is consequently faced with the problem of constantly referring to the network diagram and user's manual. This is very time consuming and confusing. Second, network information has to be divided due to the data-input limitation and, as a result, some input information is duplicated. This also requires the coder to recall prior input data, a situation that in many cases leads to inconsistencies. Finally, option spaces have to be provided within the input-data field in order to accommodate a variety of situations. Such option spaces are scattered throughout the input-data field and may not follow any apparent logic from the user's point of view.

Because of the above-stated conditions, many errors may occur in the input-data file. The NETSIM preprocessor has the capability to check the consistency of the input data and inform the user of such inconsistencies via error messages. However, in some cases error messages are given in terms of the link numbers assigned by NETSIM according to the order in which the links were input on the link-geometry cards. It is likely that the coder is unaware of the link-number assignment until faced with an error message. As a result, decoding time is spent in relating the error messages to the links on the link-node diagram. Problems such as the one described above can be classified as data-debugging problems.

NETSIM has the capability of generating printed measures of effectiveness (MOEs) either at predetermined times or at the end of the simulation subinterval and performing statistical analysis on multiple subinterval runs. The printouts may be voluminous and, although they are presented in an appealing format, they are sometimes difficult to interpret. Although the outputs are useful in defining the existence of potential problems, it may be difficult for the user to understand how such problems have evolved during the simulation time interval. Following through all the intermediate outputs is an impossible task.

With regard to the three problem elements associated with the use of the NETSIM program, interactive computer graphics (ICG) can aid in reducing or even eliminating many of the difficulties. The use of graphic output displays from computer programs is becoming common. Many computer programs are now being written in such a manner that the user can follow elements of the program execution (dynamic displays) and change the program during execution (interactive programs). The utility of computer graphics in the field of transportation was initially brought into focus at a conference held in Seattle in 1973 (3). Since then, applications in this area have been numerous. An article by Schneider (4) reviewed various applications of computer graphics in the transportation field. It is believed that alone or when combined with existing packages, computer graphic routines are cost-effective tools that can be used in analyzing data, generating and evaluating alternative solutions, and presenting the results.

Previous application (5) of computer graphics in conjunction with the NETSIM program has produced animated displays of vehicle movements and signal indications on an urban street network. The display of the results of the simulation model can be used in searching for high-performance traffic management strategies. The NETSIM/ICG computer graphic program described in this paper is an attempt to enhance the capabilities of the existing NETSIM program in addition to the animation of vehicular flow. The modifications and further developments of the NETSIM computer graphics program are described. The overall framework of the NETSIM/ICG program is presented in Figure 1. The program consists of three interactive computer graphics programs: pre-NETSIM (PRENET), NETSIM display (NETDIS), and postdisplay (POSDIS). The PRENET program provides the capabilities for interactive data input and modification, data display, and preparation of the input-data files for both NETSIM and NETDIS. The NETDIS program runs in conjunction with NETSIM and provides real-time displays of link-specific MOEs generated by NETSIM and, in addition, prepares a display file for POSDIS. The POSDIS program provides passive displays of user-selected link-specific MOEs.

SYSTEM DESCRIPTION

NETSIM/ICG is a FORTRAN-based program. It was designed for use on Rensselaer Polytechnic Institute's IBM 3033 computer system that has an IBM 3277 graphics attachment and a graphics attachment support program PRPQ. The IBM 3277 graphics attachment uses a dual-screen work station concept. All alphanumeric data that are related to, but not part of, the contents of the graphic display are managed by an IBM 3277GA terminal. Graphic displays and graphical data manipulations are done on a Tektronix cathode-ray tube (CRT) that has "write-through" ability (limited refresh vectors) together with supporting interactive devices. The PRPQ is made up of a collection of FORTRAN and ASSEMBLER language subroutines that are available from the user's application program for a variety of graphical and control functions.

OBJECTIVES

The objective of the NETSIM/ICG project was to develop interactive computer graphics programs that have minimum alterations to the existing NETSIM program and the following capabilities:

1. Allow user to interactively create and/or modify link-mode diagram that represents street network;
2. Allow user to interactively input and/or modify information needed for NETSIM (efforts were made to categorize NETSIM input information into major groups and to incorporate logic that eliminates all unnecessary information requests);
3. Allow user to input all required information without reference to user's manual and in free format;
4. Provide graphical displays to as great a degree as possible of network-related NETSIM input data;
5. Create input data files necessary for both NETSIM and NETDIS from information previously input and read in these data files for modification;
6. Provide real-time displays of selected link-specified MOEs;
7. Allow user to interrupt execution, change display variables, and then resume execution of sim-

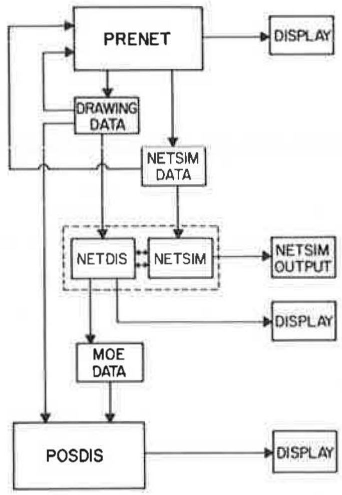


Figure 1. Overall framework of NETSIM/ICG computer program.

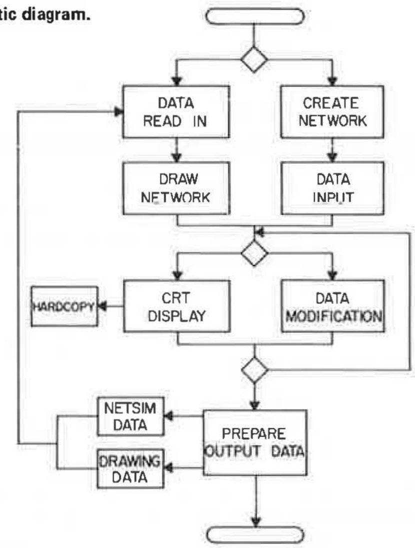


Figure 2. PRENET schematic diagram.

ulation and to prematurely terminate execution without losing standard cumulative statistical outputs generated by NETSIM;

8. Generate a graphical link-specific MOE data file for future passive displays; and

9. Provide passive displays of link-specific MOEs.

THE PROGRAM

Due to the size and the complexity of the NETSIM/ICG project, only portions of the features will be presented here in order to demonstrate the capability of the computer graphic aid. The description of the program is divided into three major sections. Each section addresses one of the problems identified in the problem statement, namely, input-data preparation, data display (debugging aid), and output-data interpretation.

Input-Data Preparation

This capability is provided by the PRENET enhancement program. The conceptual structure of the PRENET program is presented in Figure 2.

Initially, the user is asked to type in the information concerning the title of the run, the name of the network, the city, the state, and the date. This information is displayed at the top of the CRT screen and used as an identification code for this run.

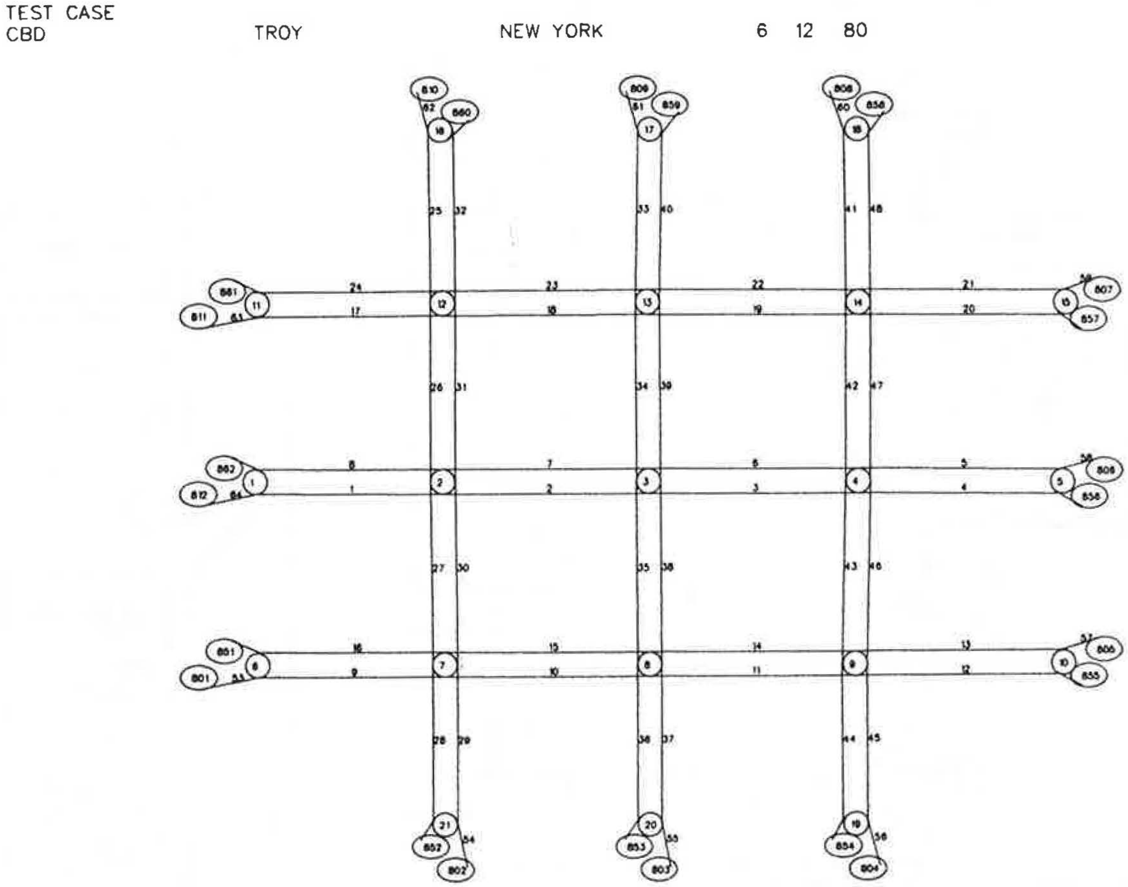
The NETSIM model uses a link-node network description of the actual system, and in order to accommodate a variety of urban street networks, the program has to have the capability of allowing the user to interactively create such a link-node diagram. Initially, the user is asked to use the cross-hair cursor to identify the location of nodes on the CRT screen. Each time a node location is identified, a circle that has an identifying node number is drawn on the screen (the circle is drawn within a subpicture for future subpicture detection). During the node-input stage the user has the capability of changing or deleting an already defined node. After the user exits from this input stage, the node numbers and their coordinates are fixed and cannot be changed. Second, the user must identify the upstream and downstream nodes of each link with the cross-hair cursor. Each time a link is thus identified, a line is drawn that connects the nodes and a link number appears on the screen adjacent to the link. As in the node-input stage, the user has the option to change or delete any link; thereafter, the links remain fixed. After the user has input all the nodes and links (NETSIM classifies nodes and links as entry nodes, exit nodes, internal nodes, exit links, entry links, and internal links), the complete link-node diagram is drawn and pertinent information is stored in appropriate arrays for later use. Such a link-node diagram is illustrated in Figure 3.

The NETSIM input data can be categorized into six major groups: link information, signal information, input traffic volumes, bus information, short-term and long-term events, and simulation control information. The program prompts the user for input and, if necessary, provides a short description of the appropriate code. Certain displays appear on the link-node diagram at appropriate times to aid the user. For example, the particular link for which input is being prompted is displayed in the flashing mode. Such a display is very helpful in identifying destination nodes that receive left-turning, through, and right-turning vehicles. Precautions have been taken in the program to reject obviously erroneous input data. For example, NETSIM requires that a bus route always start at an entry node. If the first node input for a bus route is not an entry node, the program will print an error message and prompt for corrected input. Other options, such as redoing portions of the input data, have also been provided for so that apparent mistakes can be corrected immediately. These options are provided in various places in the program, including the end of each link-information input, phase-information input, bus-station information input, bus-route information input, etc.

An effort has been made to organize the input-information procedure according to its logical flow. For example, during the link-information input stage, the program will prompt the user for through-traffic information if no left-turn destination node is designated. Questions concerning the left-turn percentage and left-turn pocket capacity will be bypassed. The logical structure of the input procedures for link and bus information is outlined in Figure 4.

In many cases, the information needed by NETSIM is graphically fed directly from the CRT screen. To illustrate, consider the bus-information input, which consists of bus station information and bus route information. For bus station information, the link on which the station is located is identified on the link-node diagram by the cross-hair cursor.

Figure 3. Sample link-node diagram.



Other alphanumeric data such as lanes blocked, type of station, dwell times, etc., are typed in by means of the keyboard. At this point, a rectangle, which represents the station, is drawn. Adjacent to the rectangle are shown the station number and dwell time (Figure 5). For bus route information, the links are erased and only the nodes and bus stations are displayed on the screen (Figure 6). For each bus route, the user must identify a series of succeeding nodes that connect this bus route, beginning with an entry node and terminating at an exit node. Concurrently, a line is drawn between the nodes each time a new node is identified (Figure 7). After having defined a particular bus route, the user is asked to identify the bus stations associated with this route by using the cross-hair cursor. A circle is drawn around each of the bus stations as they are identified (Figure 8). This is very useful in indicating data that have already been entered. Finally, the user types in the bus headway for this route. The above options, provided by PRENET, illustrate how the user, working on the link-node diagram, can input the bus information interactively.

Similar procedures are used in the input of signal information. For each phase, the user works with a signal-phasing diagram, which is displayed on the screen. Variable initial-green and gap-reduction information for actuated signals is displayed on top of the phasing diagrams, if applicable. Detectors are drawn and identified (the letters I and A are used to identify inactive and active detectors, respectively) on the "lane-detailed network plot" on the appropriate link and at the appropriate distance from the stop line. The phase numbers that are serviced by the detectors are also indicated on the screen. The user, in effect, puts the detectors on each approach at an actuated intersection (Figure 9). The data input logic is such that separate surveillance information is not required.

Currently, the interactive data input portion of the PRENET enhancement program cannot process optional input information such as link names, embedded data changes, and subinterval update data.

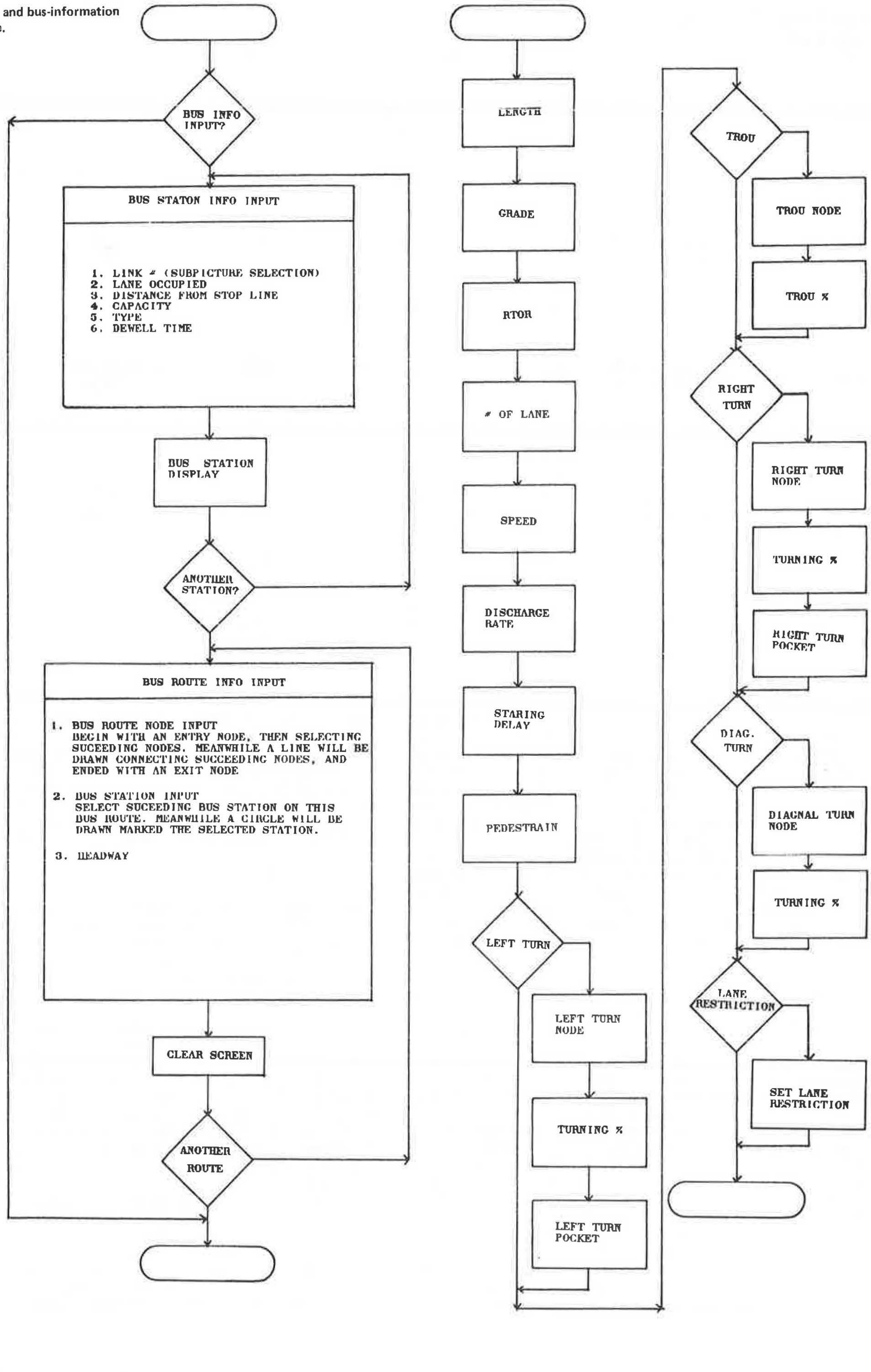
Data Display

After the necessary information for NETSIM and NETDIS is obtained by using the interactive data input capabilities provided by PRENET, the information is written into data files in a format that conforms to the NETSIM data input requirements. The capability of displaying these data is also provided by PRENET.

Since NETSIM is actually modeled on a lane-detailed microscopic network and keeps a record of each vehicle in each lane on the links of the network, it is desirable for the user to have a graphic display of such a lane-detailed microscopic network. A schematic lane-detailed microscopic network plot serves that purpose. An algorithm has been developed to produce such a plot. The word "schematic" is used because the length of the link and the width of the lane are not of the same scale. An example of such a schematic lane-detailed microscopic network diagram produced by this algorithm is presented in Figure 10. This schematic lane-detailed network plot is useful in displaying lane-specific information such as turning pockets, lane channelization, signal detectors, etc.

In coding the existing NETSIM signal information,

Figure 4. Link- and bus-information input procedure.



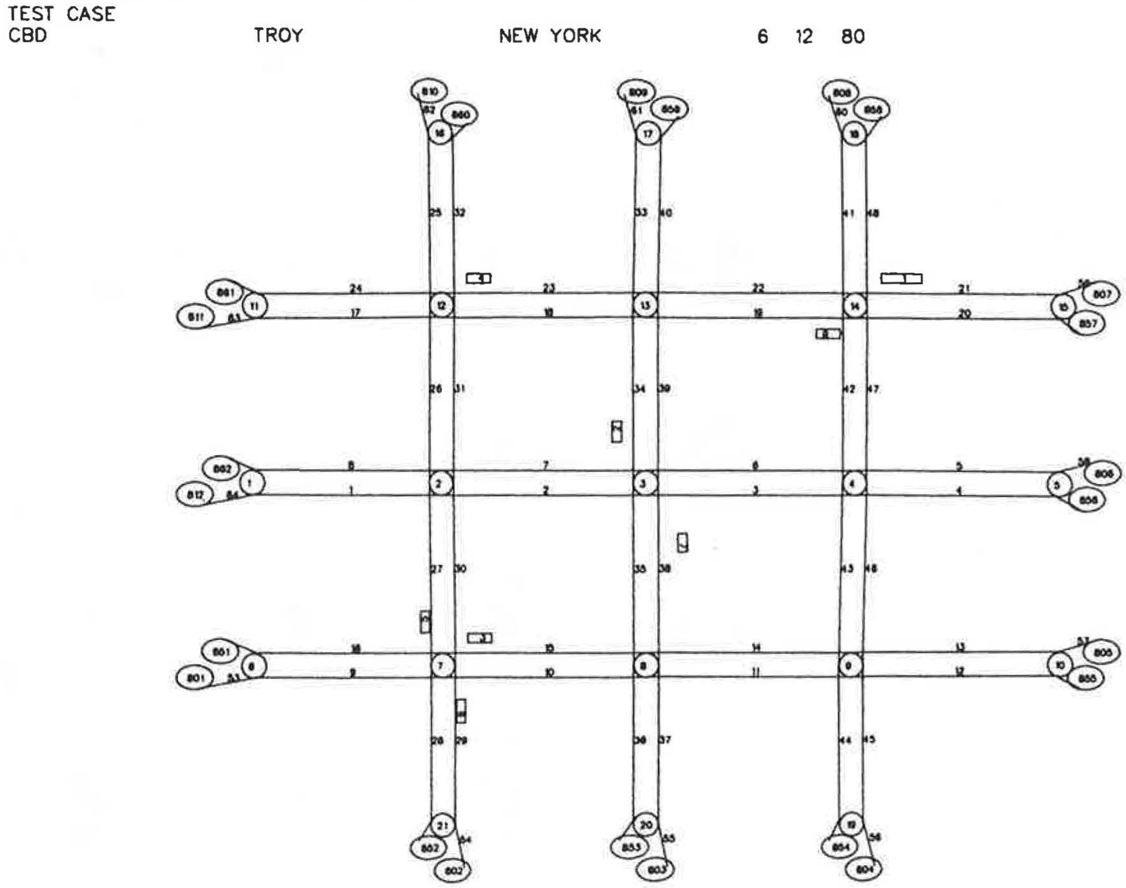


Figure 5. Bus station display.

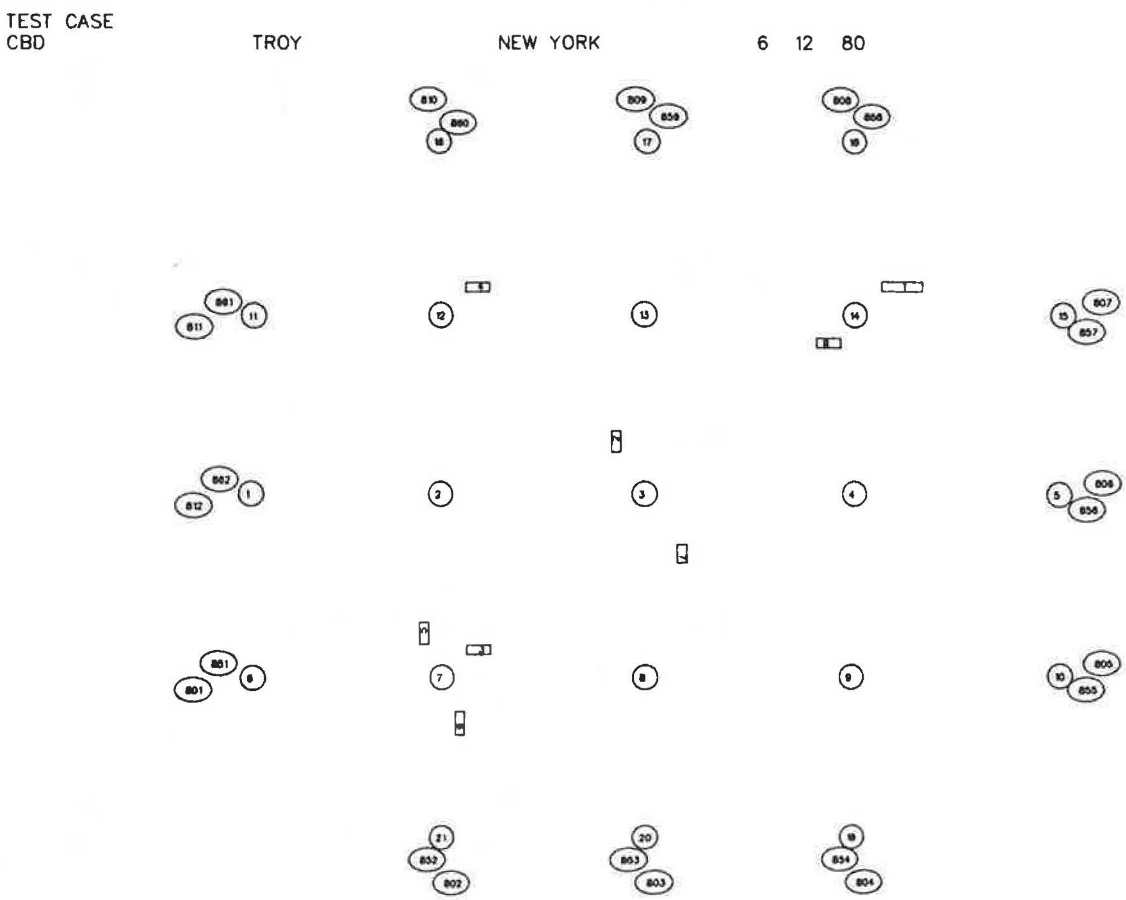


Figure 6. Bus station and node diagram.

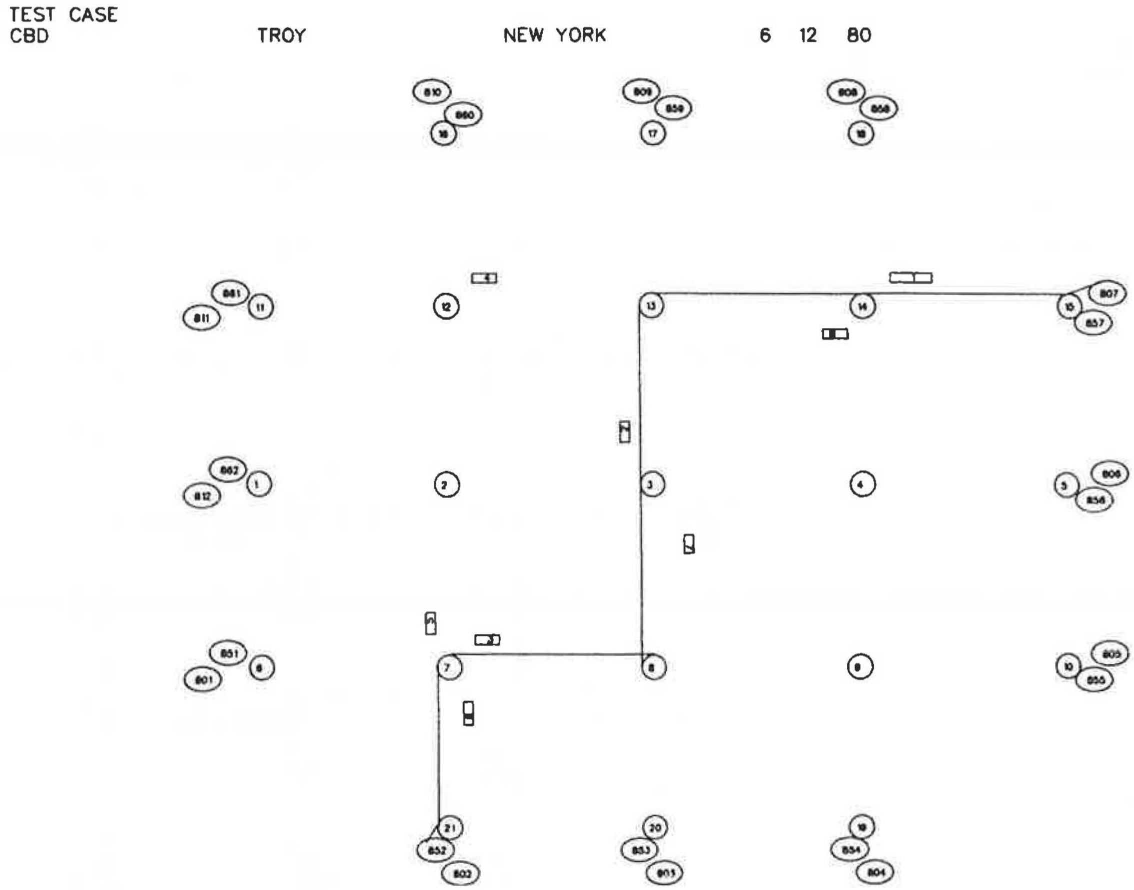


Figure 7. Bus route display.

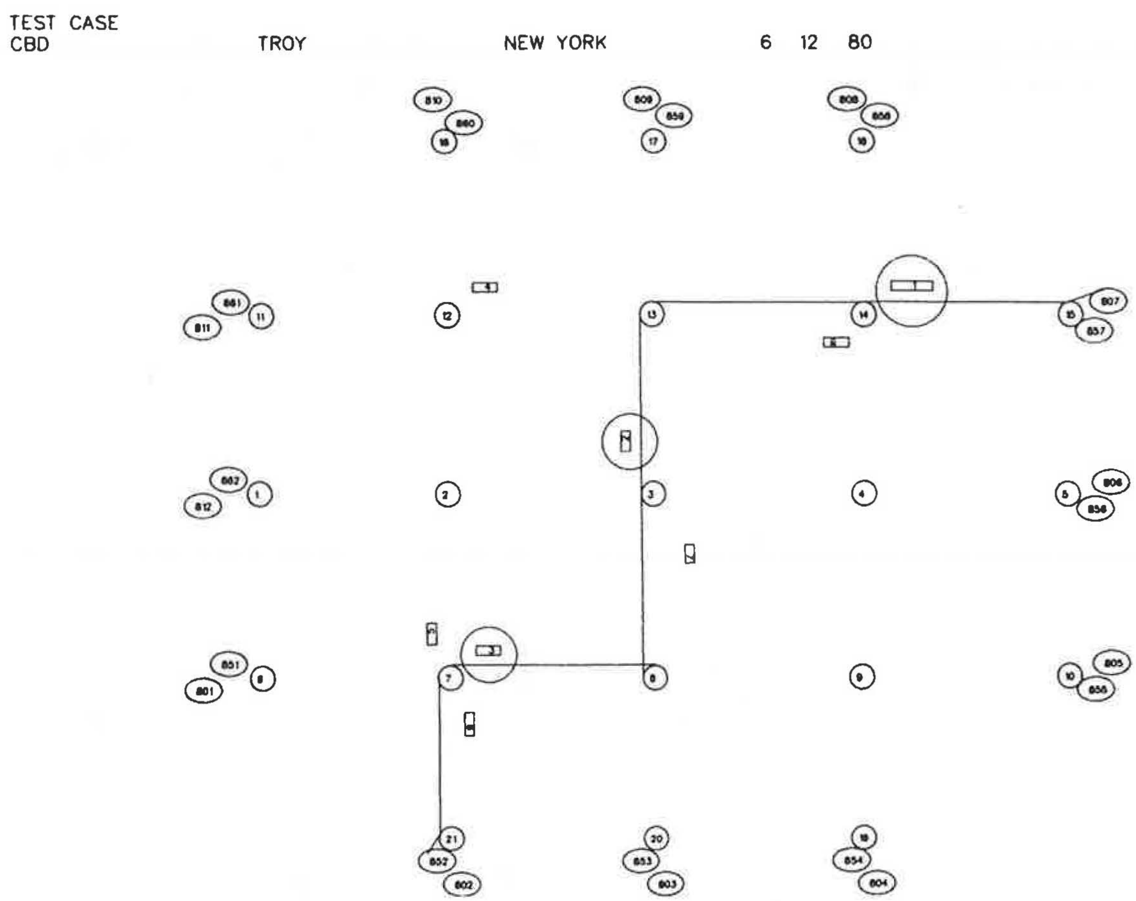


Figure 8. Bus station display.

Figure 9. Schematic lane-specific network drawing.

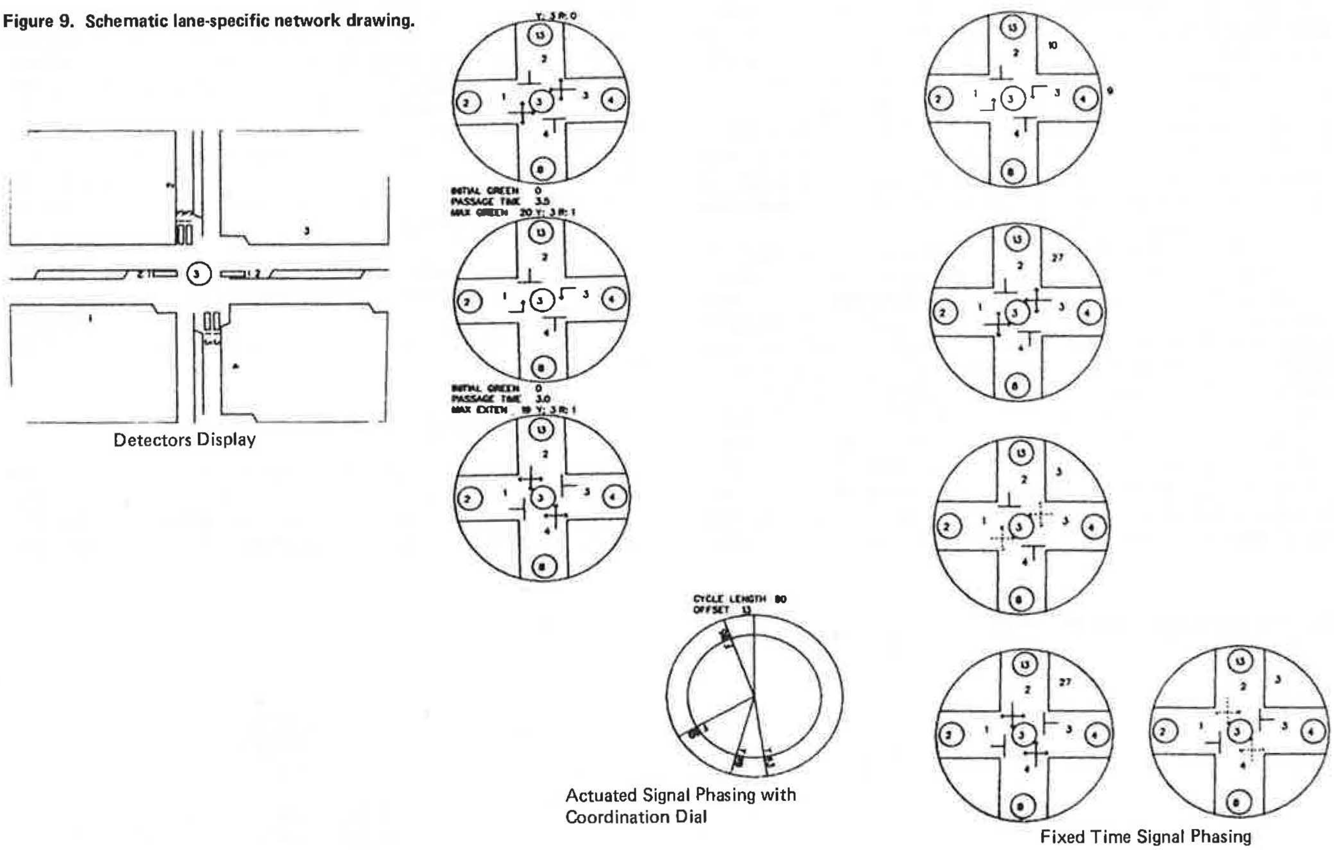
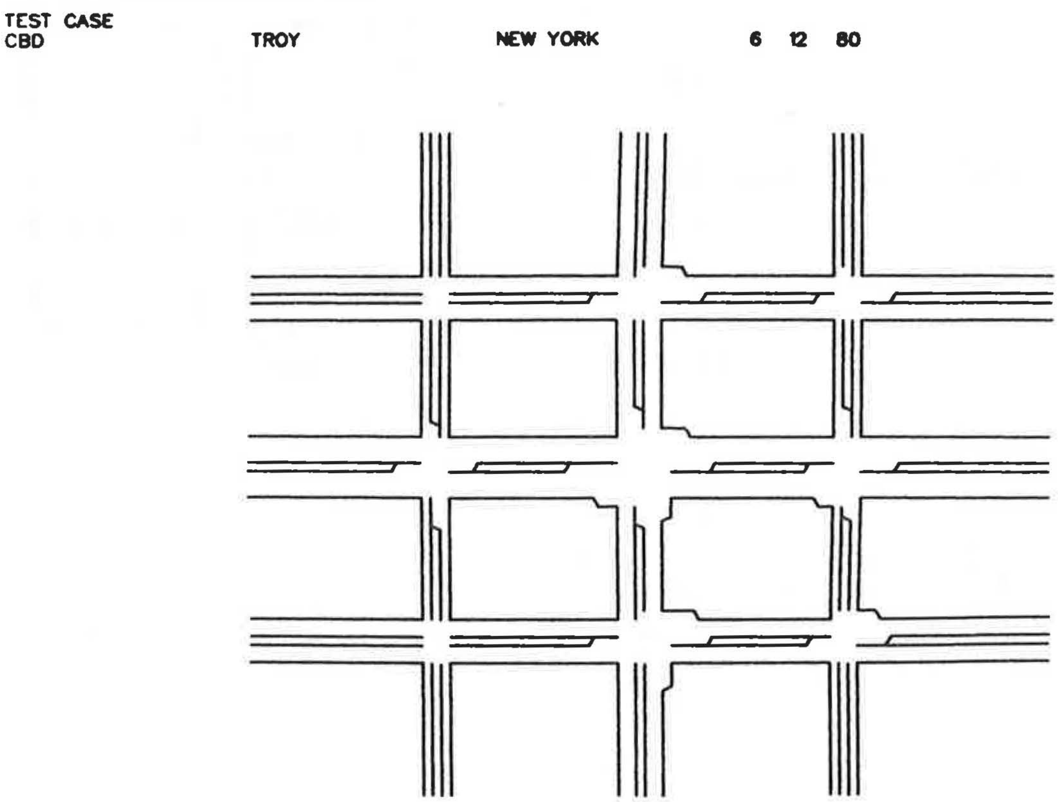


Figure 10. Link-specific information display.



the user is constantly dealing with approach numbers and signal codes. It is desirable for the user to visualize the signal operation by providing signal-phasing diagrams. PRENET has the capability of providing up to nine different phasing diagrams along the left and bottom edges of the CRT screen. For the actuated signals, controller timing and detectors are also displayed. Furthermore, a plot of the coordination dial is provided if the actuated signal is coordinated.

PRENET has the capability of allowing the user to interactively display the NETSIM input data by means of menu selection. For link information, the user can display (a) the schematic lane-detailed diagram with turn-pocket capabilities, channelization, link length, and free-flow speed (Figure 11); (b) destination nodes that receive the left-turn, through, and right-turn vehicles (Figure 11); (c) turning percentages (Figure 11); and (d) other miscellaneous information such as right-turn-on-red code, pedestrian code, grade code, queue-discharge rate, and starting delay (Figure 11). For signal information, only one node can be displayed at a time (Figure 9). The user can have all detectors within the network displayed. For input traffic volumes, only one display is needed (Figure 11). For bus information, the user can display (a) all bus stations on a schematic lane-detailed network (Figure 11) and/or (b) either all or some bus routes and their associated bus stations (Figure 12). For event information, the user can either display both short-term and long-term events in one plot (Figure 11) or can display them separately.

PRENET has the capability of allowing the user to interactively modify an already defined data base by using menu selection. The geometry of the link-node diagram cannot be modified. The user can delete, change, or add to the already defined NETSIM information.

Output-Data Interpretation

This section addresses the third identified problem element associated with the use of the NETSIM program, namely, the analysis of the outputs. As alluded to previously, the aggregated nature of the

Figure 11. Signal phase and detector display.

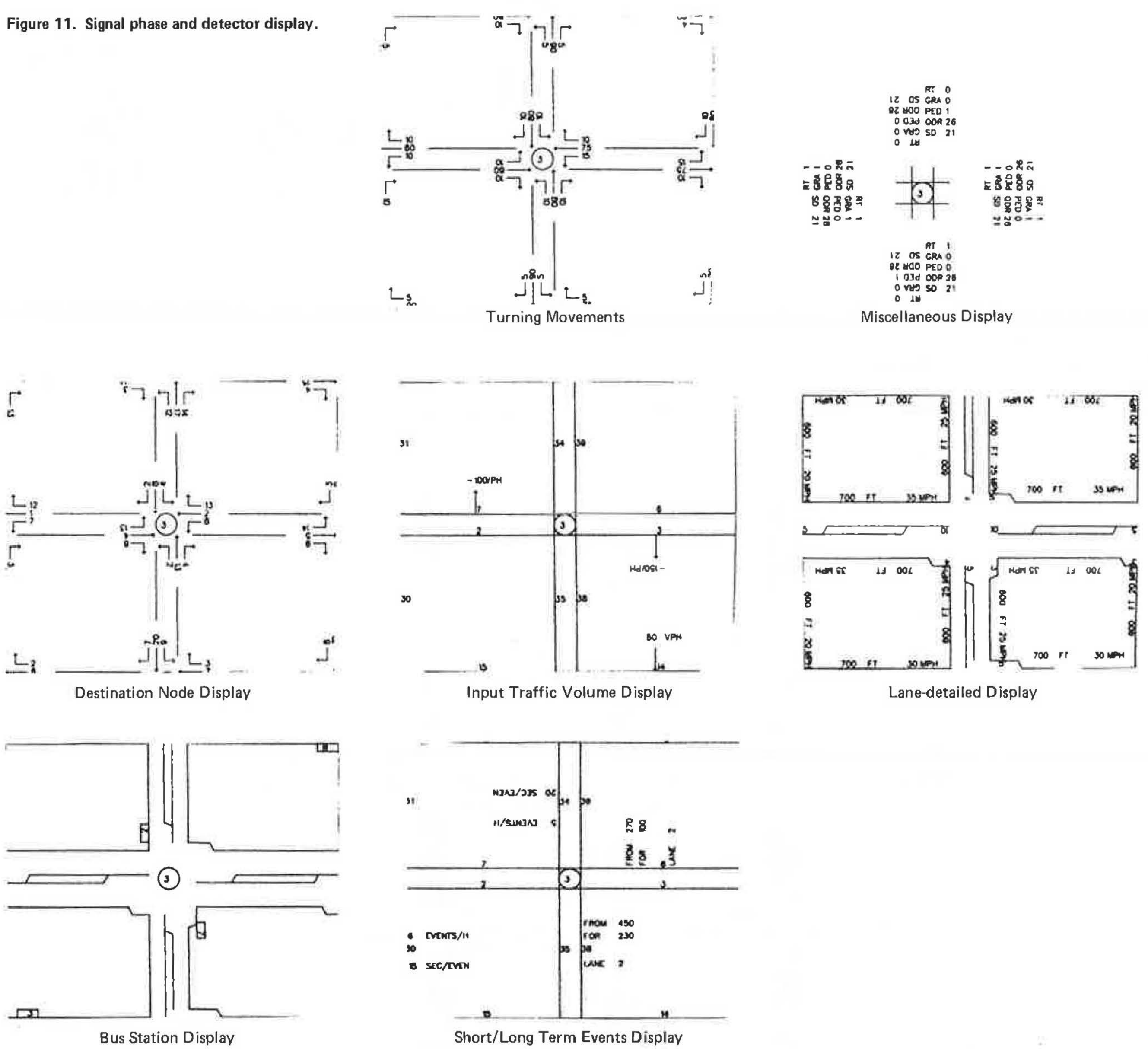
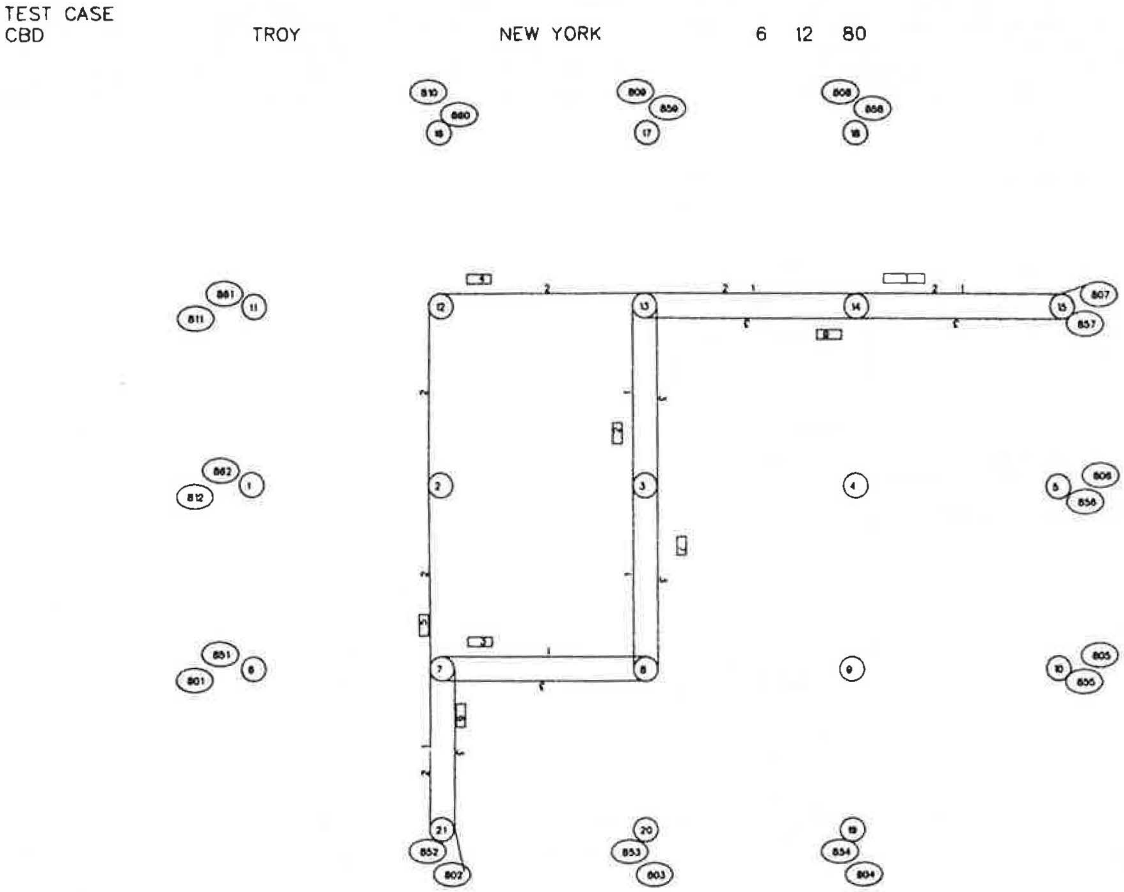


Figure 12. Bus route and bus station display.

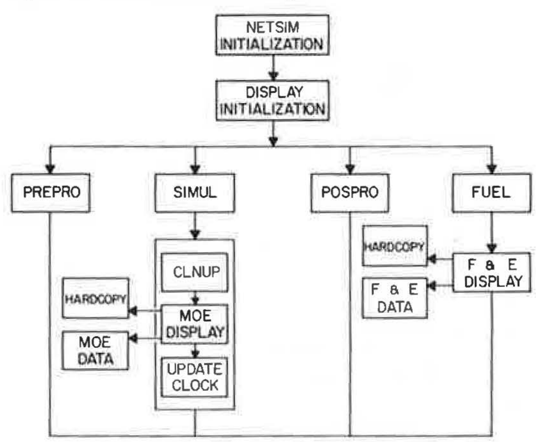


final simulation output tends to obscure much of the information that the program generates. Moreover, the analysis of the voluminous intermediate outputs is essentially infeasible. The following section discusses the concepts and approaches followed in designing the NETDIS and POSDIS enhancement programs that are a part of the developed NETSIM/ICG program. Both NETDIS and POSDIS were developed with the goal of facilitating the assimilation and interpretation of the network-simulation outputs.

Other researchers have previously applied computer graphics techniques to display the results of NETSIM. Joline (6) has produced a film that displays the movement of individual vehicles through the network. It has been useful in showing potential users of NETSIM what the model does. His work has also helped to identify errors in the model and has led to subsequent modifications. Eiger, Chin, and Woodin (5) have also developed a program that produces animated displays of the results of the simulation model that can be used in searching for high-performance traffic management strategies. Schneider, Combs, and Folsom (7-9) have taken a different approach, in which the interest is in displaying data that describe the overall performance of the system over time. By examining these system-wide displays, the user should be able to generate some ideas on modifying the parameters of the system to obtain higher levels of performance. This is a holistic approach to the problem of discovering ways to drive the performance of a multiobjective system in the desired directions.

The operational frameworks of the NETDIS and POSDIS programs are shown in Figures 13 and 14, respectively. The MOEs generated by the NETSIM

Figure 13. NETDIS schematic diagram.



program and displayed by NETDIS and POSDIS are shown in Table 1. Fuel-consumption and emissions data can be obtained only at the end of each simulation subinterval. The remaining MOEs can be obtained each simulation second. Queue length and occupancy are instantaneous displays; the others are cumulative or aggregate.

The cumulative link-specific MOEs are displayed on a three-dimensional perspective plot. The link-node network is drawn on the x-y plane, and the MOEs are represented by perpendicular lines (Figure 15) or rectangles (Figure 16) in the z-direction with the height scaled appropriately. The scale factors for all the MOEs are predetermined along with the parameters involved in generating the perspective plots. The user currently does not have the option of selecting these scale factors or perspective plot parameters. This maintains consistency among plots for different runs so that they can easily be compared. Often, portions of the selected MOE displayed on the perspective plot may be obscured by others. In other instances, the displays can be confusing due to the size of the network. Under these conditions, the user may wish to concentrate on certain areas or links of the network. The

Figure 14. POSDIS schematic diagram.

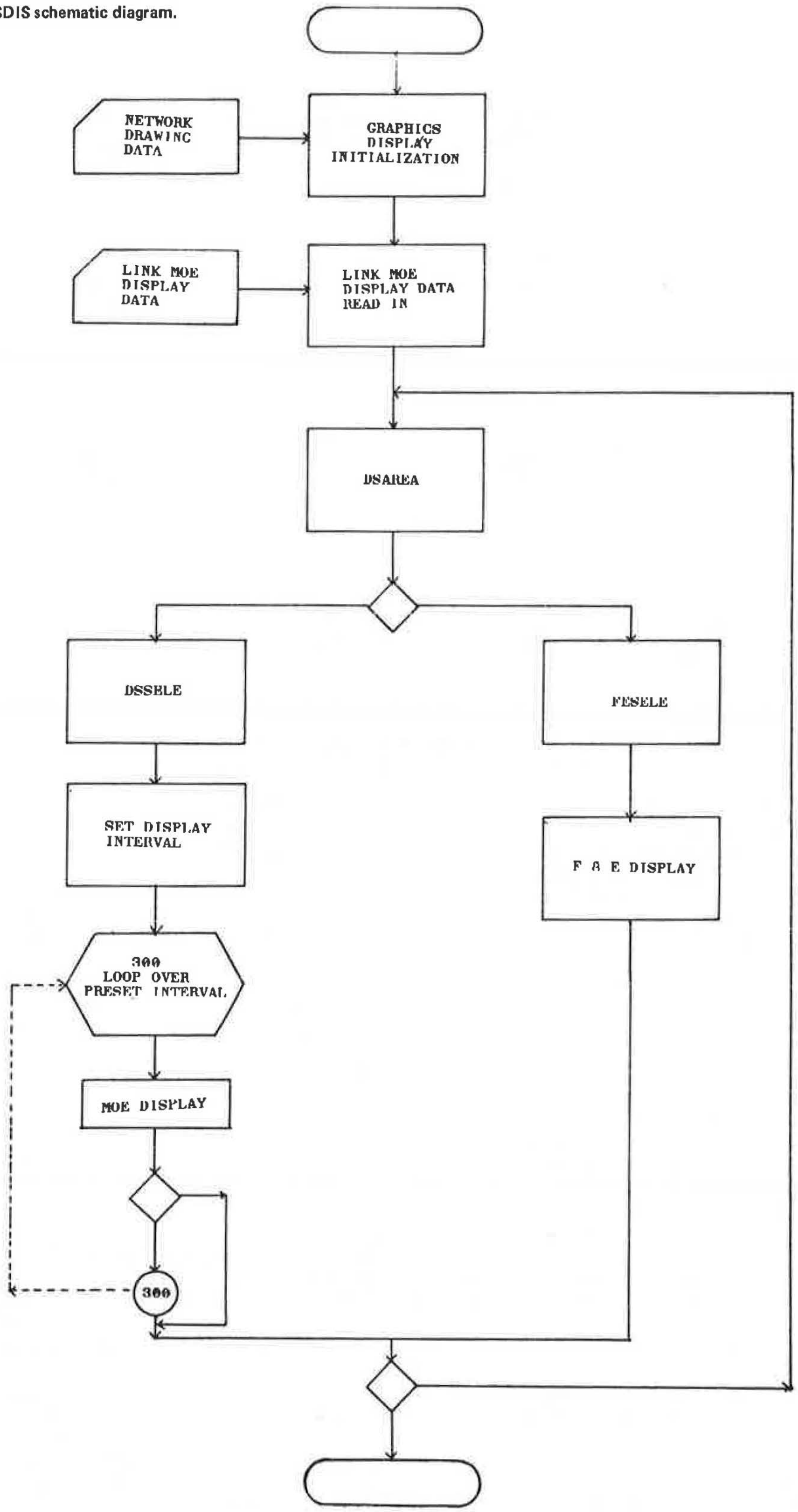


Table 1. Link-specific MOEs displayed by NETDIS and POSDIS.

Link-Specific MOE	Unit	Menu Abbreviation	Availability	Statistic	Display Plot
Queue length	Vehicles	Q-LEN	Every second	Instantaneous	2-D lane-specific
Occupancy	Vehicles	OCCU	Every second	Instantaneous	3-D perspective
Vehicle miles	Miles	V-MI	Every second	Cumulative	3-D perspective
Vehicle discharged	Trips	V-DIS	Every second	Cumulative	3-D perspective
Moving time	Minutes	MOV-T	Every second	Cumulative	3-D perspective
Delay time	Minutes	DLY-T	Every second	Cumulative	3-D perspective
Moving time/total trip time	Percent	M/T	Every second	Cumulative	3-D perspective
Total travel time	Minutes	T-TIME	Every second	Cumulative	3-D perspective
Travel time/vehicle	Seconds	T-TIM/V	Every second	Cumulative	3-D perspective
Travel time/vehicle mile	Seconds	T-TIM/V-M	Every second	Cumulative	3-D perspective
Delay time/vehicle	Seconds	DLY-T/V	Every second	Cumulative	3-D perspective
Delay time/vehicle mile	Seconds	DLY-T/V-M	Every second	Cumulative	3-D perspective
Stop delay/total delay	Percent	PCT ST-DLY	Every second	Cumulative	3-D perspective
Average speed	Mph	AV SP	Every second	Cumulative	3-D perspective
Average occupancy	Vehicles	AV OCC	Every second	Cumulative	3-D perspective
Stops/vehicle	Stops	ST/V	Every second	Cumulative	3-D perspective
Average saturation	Percent	AV SAT PCT	Every second	Cumulative	3-D perspective
Cycle failures	Failures	CYC FAL	Every second	Cumulative	3-D perspective
Fuel consumption (automobile)	Gallons	FUEL-AUTO	At end	Aggregate	3-D perspective
Fuel consumption (truck)	Gallons	FUEL-TRUCK	At end	Aggregate	3-D perspective
Fuel consumption (bus)	Gallons	FUEL-BUS	At end	Aggregate	3-D perspective
Fuel efficiency (automobile)	Mpg	MPG-AUTO	At end	Aggregate	3-D perspective
Fuel efficiency (truck)	Mpg	MPG-TRUCK	At end	Aggregate	3-D perspective
Fuel efficiency (bus)	Mpg	MPG-BUS	At end	Aggregate	3-D perspective
HC (automobile)	Grams	HC-AUTO	At end	Aggregate	3-D perspective
CO (automobile)	Grams	CO-AUTO	At end	Aggregate	3-D perspective
NO X (automobile)	Grams	NO-AUTO	At end	Aggregate	3-D perspective

NETDIS and POSDIS programs have the option of restricting the display area (Figure 17).

Queue lengths, by contrast to the cumulative MOEs, are displayed on a lane-detailed schematic network plot. The queue of a particular lane is drawn on that lane and scaled to the corresponding link length on the lane-detailed plot (Figure 18). By using these and other displays, the user can gain an intuitive understanding of the overall network operation.

Since NETDIS and POSDIS can only display one selected MOE at a time, the capability is provided to interactively select the MOE to be displayed. This capability is through menu-option selection.

As execution proceeds, the user has the option to change the selected MOE or the display area or to terminate the simulation prematurely without losing the link-performance statistics accumulated to that point. These options are provided by using the ATTENTION-TRAP feature inserted at the end of the NETSIM simulation control loop.

In displaying the selected MOEs, the following user-selected options have been provided:

1. The selected MOE can be displayed in either the storage or the refresh vector mode.
2. The user can choose either lines or rectangles to represent the magnitude of the MOE (Figures 15 and 16).
3. The selected MOE can be displayed with or without numerical values of its magnitude (Figures 15, 16, and 17).
4. The user can obtain hard copies of all displays.
5. The user can write all the MOE graphic data into a data file for future passive displays.

After the NETSIM simulation finishes executing through the predetermined simulation time subinterval and the NETSIM subroutine FUEL has been executed, the user can display selected link fuel and emissions data (Figures 15 and 16).

The user may wish to examine the link-specific MOE displays more than once. This is particularly true since, as is recalled, NETDIS can only display one MOE at a time. Instead of resimulating, the user can call a separate program (POSDIS), which can repetitively display the MOE data generated by NETDIS. The options described above can help the user gain a full understanding of the overall operation of the network over the simulation subinterval without resimulating.

IMPLEMENTATION

Further work in this area would address the problem of portability of the NETSIM/ICG computer program. Although efforts have been made to minimize the use of system-dependent subroutines, modifications are still needed in implementing the NETSIM/ICG enhancement on other graphics systems. In general, the following functional capabilities on other graphics systems are needed to accommodate the NETSIM/ICG program without major modifications:

1. MOVE to initial point;
2. DRAW line from initial point;
3. Ability to create subpicture (entities, cells);
4. Device to interactively pick subpictures on CRT screen;
5. Device to interactively pick x-y location input on CRT screen;
6. ATTENTION-TRAP feature; and
7. Graphics processor that has capability to handle large network.

Other software subroutines such as circle drawing, ellipse drawing, perspective drawing, intersection of two linear angles of line, etc., are supported by the NETSIM/ICG program.

SUMMARY AND CONCLUSIONS

The NETSIM program is a useful tool in evaluating alternative traffic-control strategies. However, the numerous and highly complex initial data preparation tasks significantly increase the costs associated with using the program. The interactive data input elements of the PRENET enhancement program demonstrate the potential usefulness of interactive data input for both NETSIM and its other enhancements, NETDIS and POSDIS. The program provides a systematic method for obtaining the information needed by NETSIM without reference to the user's manual or particular attention to correct input-data

Figure 15. Perspective plot with line display.

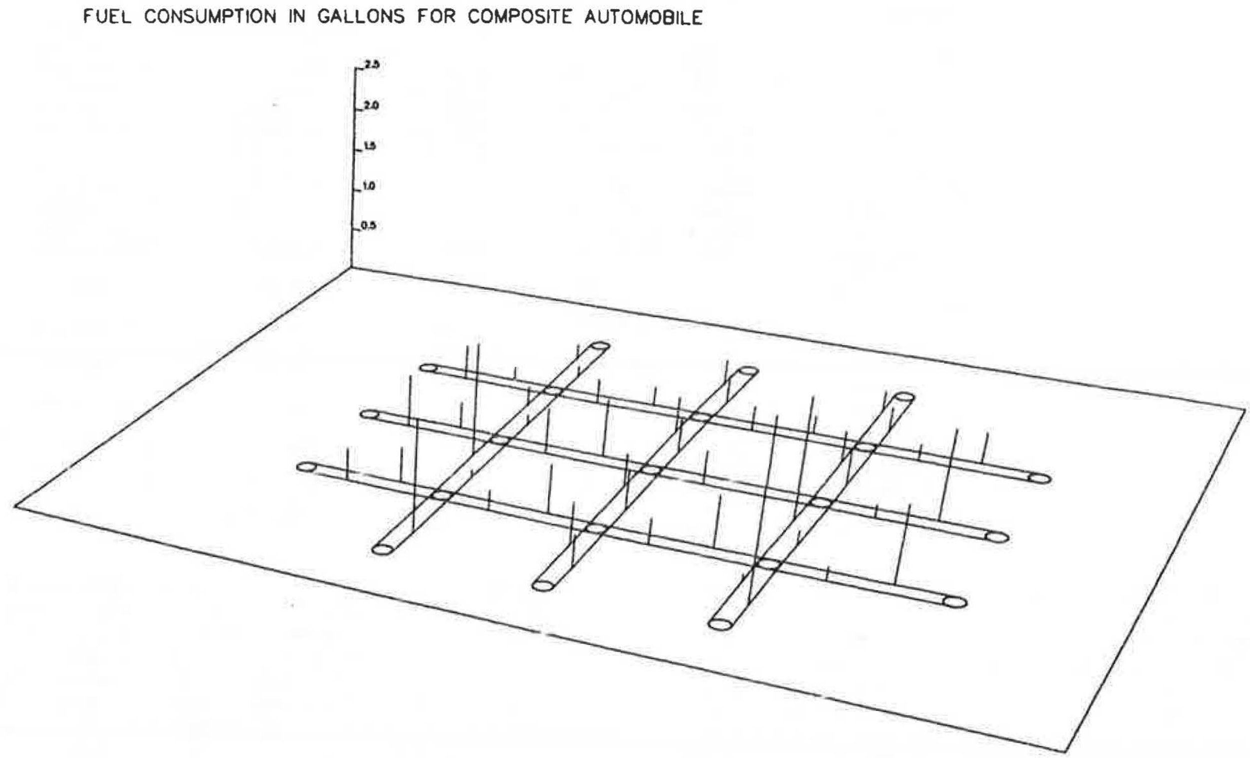
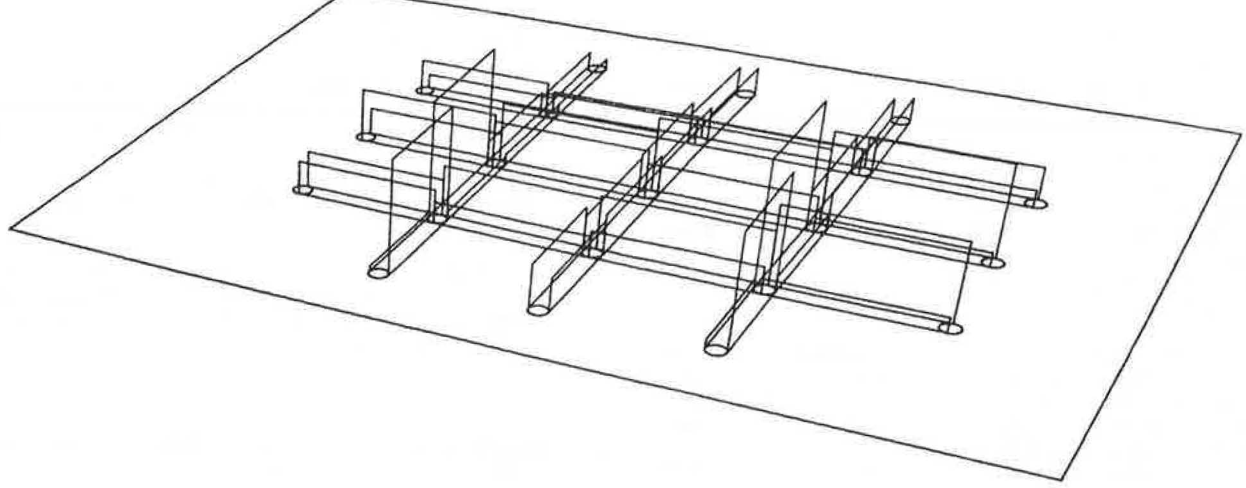


Figure 16. Perspective plot with rectangle display.



fields and codes. In addition, certain information is input graphically and directly on the CRT screen. The capabilities provided by PRENET permit the processing of complex situations with minimal data input problems. In addition, the program has been designed so that terminal sessions can be interrupted without loss of the previously input data. Providing for interrupted terminal session capability is extremely important in an interactive data input process.

The microscopic nature of the NETSIM simulation model requires the user to prepare numerous input data, a process that can be tedious. Very often, inconsistencies, contradictions, or other errors result in the input-data file. The interactive computer graphics program PRENET, by displaying the NETSIM input data, provides the user with an intuitive understanding of the data. Errors within the input data file will produce either an erroneous network display or obvious inconsistencies on a net-

Figure 17. Partial perspective plot with rectangle display.

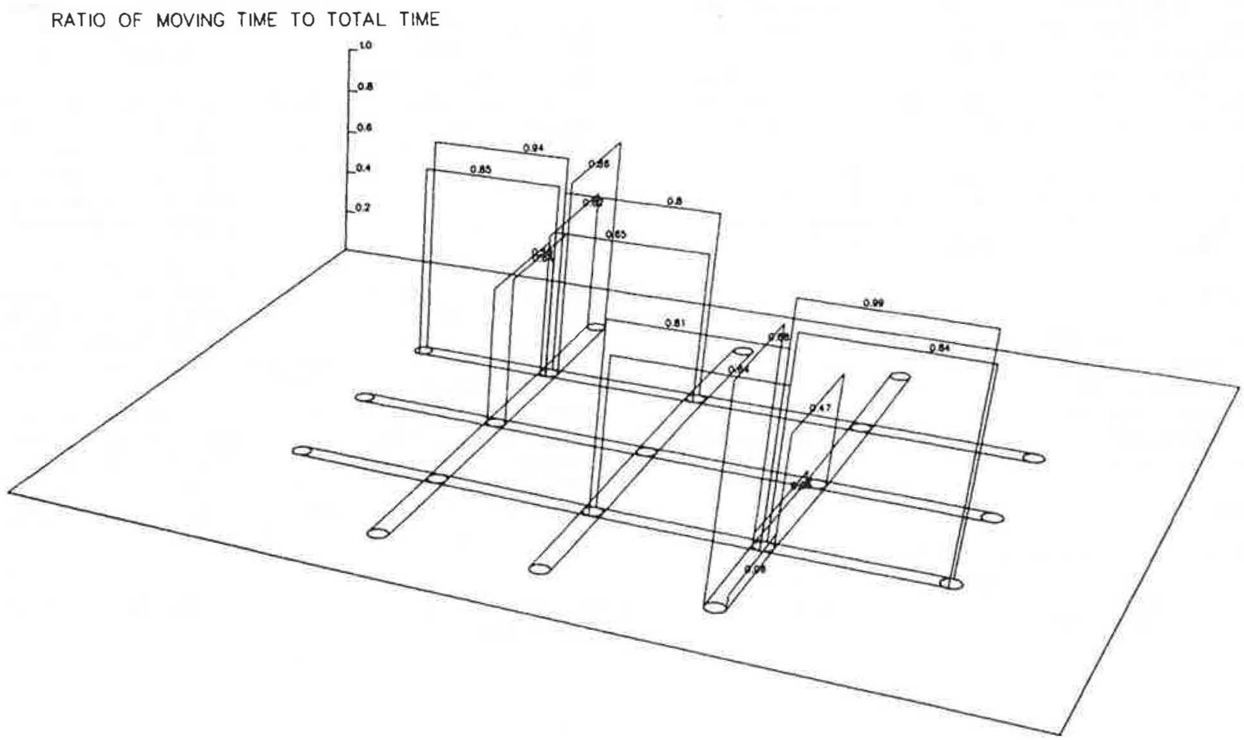
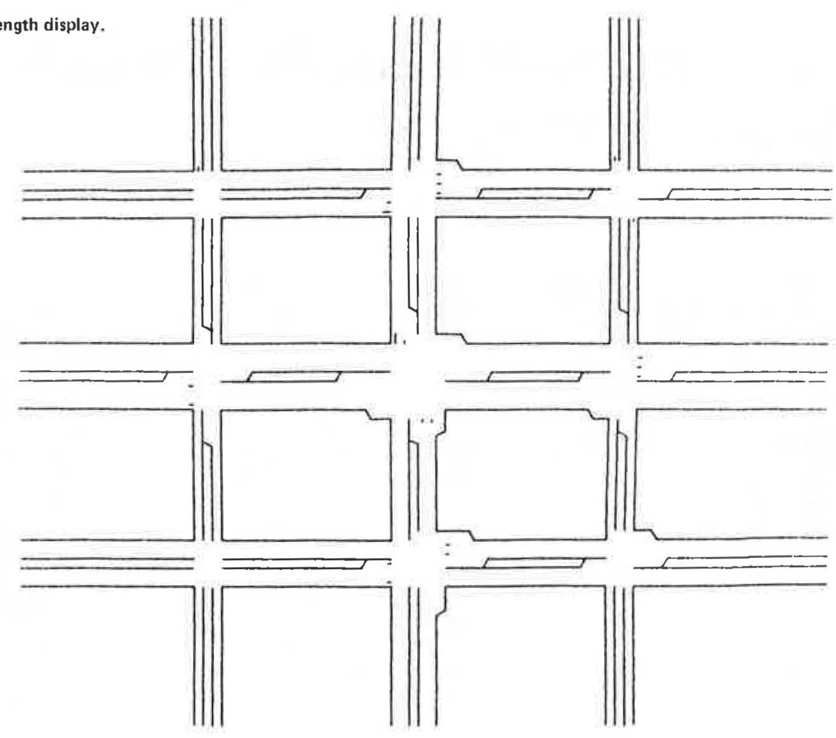


Figure 18. Queue-length display.



work diagram. Subsequently, these errors can be corrected by using the interactive data modification capability provided. This portion of the PRENET enhancement provides the user with an intuitive, direct, and efficient method to debug the NETSIM input data.

Having used NETSIM to define potential problems that relate to traffic control strategies in an urban street network, traffic engineers must gain a full understanding of how and why such problems have evolved. The ICG enhancements NETDIS and POSDIS demonstrate the feasibility and usefulness of graphically displaying link-specific MOEs as generated by the NETSIM simulation model. Such displays provide the user with more easily assimilated information with which operation of the network can be comprehended. More precisely, the lane-specific queue build-ups can be visualized, and the overall network relative performance measures can be obtained at a glance.

With respect to the development of additional graphics capabilities, work is required to generate time-space diagrams and displays of signal indications on the lane-detailed network plots.

ACKNOWLEDGMENT

This project was partly funded by the Research and Special Programs Administration of the U.S. Department of Transportation. The results and views expressed herein are ours and do not necessarily reflect the policies or views of the U.S. Department of Transportation.

REFERENCES

1. L. Kubel, G. Bloodgood, F. Workmon, and D. Gibson. What Network Simulation (NETSIM) Can Do for the Traffic Engineer. Public Roads, Vol. 41, No. 4, March 1978.
2. W.D. Labrum, R.M. Farr, W.J. Kennedy, and D. Gibson. Analyzing Intersection Performance with NETSIM. Public Roads, Vol. 42, No. 1, June 1978.
3. J.B. Schneider. Interactive Graphics in Transportation System Planning and Design. Final Rept. DOT-TST-74-10, Jan. 1974.
4. J.B. Schneider. Applications of Computer Graphics in the Transportation Field. Computer Graphics, Vol. 1, No. 13, 1978.
5. A. Eiger, S.M. Chin, and D. Woodin. Computer Graphic Animation of UTCS-1/NETSIM Traffic Flow. TRB, Transportation Research Record 729, 1979, pp. 7-10.
6. E.S. Joline. Computer-Generated Films to Document and Demonstrate Transportation Simulation Models. TRB, Transportation Research Record 619, 1976, pp. 34-37.
7. J.B. Schneider, D.M. Combs, and T.C. Folsom. NETGRAF: A Computer Graphics Aid to the Operation and Interpretation of NETSIM, a Traffic Simulation Model: Part 1--Overview and Experimental Results. U.S. Department of Transportation, Univ. Res. Rept. DOT-RSPA-DPB-50/80/10, 1980.
8. D.M. Combs, R.C. Folsom, and J.B. Schneider. NETGRAF: A Computer Graphics Aid to the Operation and Interpretation of NETSIM, a Traffic Simulation Model: Part 2--User's Guide. U.S. Department of Transportation, Univ. Res. Rept. DOT-RSPA-DPB-50/80/11, 1980.
9. D.M. Combs and J.B. Schneider. NETGRAF: A Computer Graphics Aid to the Operation and Interpretation of NETSIM, a Traffic Simulation Model: Part 3--Programmer's Guide. U.S. Department of Transportation, Univ. Res. Rept., DOT-RSPA-DPB-50/80/12, 1980.

Publication of this paper sponsored by Committee on Computer Graphics and Interactive Computing.

Transferability and Analysis of Prediction Errors in Mode-Choice Models for Work Trips

YOUSSEF DEHGHANI, BRENDA KOUSHESHI, ROBERT SIEVERT, AND THOMAS McKEARNEY

Some analyses of predictive accuracy and transferability of disaggregate work-trip mode-choice models are reported. The prediction error is separated into three components: model error, aggregation error, and transfer error. The results show that the weighted root mean square of total error is between 25 and 60 percent of the predicted shares and the distribution of total error between error sources seems to depend on how well the model is transferred. The main results of the research are that (a) total forecasting errors may be large, especially if the model transfers poorly; (b) transferability between cities in which the transit shares are very different is poor; (c) market segmentation improves forecasting accuracy only marginally, if at all, and; (d) the type of level-of-service data, i.e., manually coded versus network based, used in model estimation and prediction has some bearing on forecasting accuracy, and the use of zonally averaged socioeconomic attributes appeared to be somewhat detrimental to prediction. These and other results are to be held tentative for reasons discussed in detail.

The sources of the total error for work-trip mode-choice models are identified and their contributions are analyzed separately. In addition, citywide predictions of travel demands are also investigated. Four data sets were used in this study. These were the data from the Minneapolis-St. Paul area (collected in 1970); the two urban travel demand forecasting surveys from the San Francisco Bay area conducted before and after the introduction of Bay Area Rapid Transit (BART) service (collected in 1972 and 1975, respectively); and the Baltimore travel demand data set, a comprehensive set of information that describes travel behavior of 967 households in Baltimore, Maryland (collected in 1977).

The effect of market segmentation on forecasting accuracy is studied by using the same model specification as that for the unsegmented market. Three types of market segments were used: households that had one car versus those that had two or more; commuters bound for the central business district (CBD) versus others; and low-income versus high-income households (annual household incomes of $12 000,