changes over time, a useful portrayal is through a computer-generated movie. Such a movie has been produced; two frames are shown in Figure 11. The example illustrated is a 45° intersection. The contours of the bivariate probability density function of along-track and cross-track errors are shown as ellipses about the planned position of each aircraft.

As the movie evolves, the ellipses move at a constant velocity through the intersection. As the first aircraft nears the center of the intersection the probability of overlap increases, as illustrated by the rising thermometer on the left. At the same time, a plot of probability versus time is created. The movie has been found to be particularly effective in explaining the general concepts of the model to a nontechnical audience, which is crucial if the analytical results are to be accepted.

CONCLUSIONS

Computer graphics has played a central role in the analysis of data and development of methodology for determining adequate jet-route separation standards. Development of a clean data base, summary description of navigational performance, estimation of lateral-overlap probability, and dissemination of the model have all been greatly enhanced by the availability of computer graphics. The use of graphical tools throughout the study resulted in substantial savings of analytical resources and an understanding of the problem that would otherwise have been very difficult to achieve.

REFERENCES

1. A. E. Beaton, Jr., and J. W. Tukey. The Fitting of Power Series, Meaning Polynomials, Illustrated on Band Spectroscopic Data. Presented at 133rd Annual Meeting of the American Statistical Association, New York, 1973.
2. P. F. Velleman. Robust Non-Linear Data Smoothers: Theory, Definitions, and Applications. Princeton Univ., Princeton, NJ, Ph.D. dissertation, June 1975.
3. J. D. Van Ness. Non-Linear Smoothing of Long-Tailed Aircraft Radar Data. Department of Civil Engineering, Princeton Univ., Princeton, NJ, M.E. thesis, 1975.
4. D. A. Livingston and E. H. Stevens. Robust, Non-Linear Smoothing of Noisy Time Series: VOR-Defined Air Route Separation Study. National Aviation Facilities Experimental Center, Federal Aviation Administration, U.S. Department of Transportation, Tech. Letter Rept. NA-78-72-L.R., 1978.
5. J. W. Tukey. Exploratory Data Analysis. Addison-Wesley, Reading, MA, 1977.
6. Methodology for the Derivation of Separation Minima Applied to the Spacing Between Parallel Tracks on ATS Route Structures. International Civil Aviation Organization, Montreal, PQ, Canada, ICAO Circular 120-AN/89/2, 1976.
7. N. W. Polhemus. Estimation of Lateral Overlap Probabilities for VOR-Defined Jet Routes. Department of Civil Engineering, Princeton Univ., Princeton, NJ, Tech. Rept., Jan. 1980.
8. G. E. P. Box and G. C. Tiao. Bayesian Inference in Statistics. Addison-Wesley, Reading, MA, 1973.
9. D. Livingston and N. W. Polhemus. Statistical Modeling of Aircraft Collision Risk for High-Altitude Jet Routes. Department of Civil Engineering, Princeton Univ., Princeton, NJ, Tech. Rept. 79-TR-19, Aug. 1979.

Applications of Interactive Graphics in Michigan's Statewide Transportation Modeling System

TERRY GOTTS AND RICHARD ESCH

Modern transportation agencies face increasing responsibilities and decreasing revenues. The only viable solution is increased productivity and efficiency. NETEDIT, an operational interactive method of updating, displaying, and interrogating networks, has allowed Michigan's Department of Transportation to increase its analysis capabilities without sacrificing production time. The utility of such a process is easily seen by contrasting manual coding and batch updating with the interactive update process. NETEDIT gives the user instant pictorial feedback, which enables correction of errors on the spot. This allows the average elapsed time for generation of alternatives to drop from two weeks to 4 h. In addition, the ability to vary the way a link is drawn—solid, dotted, dashed, crosshatched, even varying bandwidths—based on its attributes lends versatility. The addition of a digitizer tablet has allowed creation of 2000-node networks in two weeks. A tree-plotting subroutine eliminates much of the elapsed time in network calibration. Most importantly, NETEDIT, together with Michigan's Statewide Transportation Modeling System, has been used in more than 350 actual planning applications over the past two and one-half years. As a result, NETEDIT has become a valuable transportation planning technique in Michigan.

Governments face a challenge in meeting modern political realities. On the one hand, legislation and public expectations demand thorough planning and implementation of an expanding range of governmental responsibilities. On the other hand, the economic realities of the foreseeable future dictate a governmental belt-tightening; personnel and budgets will be maintained or cut back rather than increased.

Michigan's Department of Transportation (MDOT) is meeting just such a challenge in its Bureau of Transportation Planning. Although budget and personnel are not rapidly expanding, the demands on the planning process from federal legislation, MDOT's action plan, and concerned citizens have grown tremendously. MDOT must now consider many alternatives and modes and must examine an ever-widening range of impacts--travel, social, economic, and environmental--for every major transportation project it undertakes. Moreover, the people who live in the region of the proposed project and who will therefore be most affected by the final solution must be involved in each step. Based on the evaluation of the first set of alternatives, new alternatives may

Figure 1. CRT, digitizer tablet, and hard copy.

be proposed and the whole cycle repeated until a consensus is reached.

With conventional methods of network updating, interrogation, and display, the emphasis on public involvement becomes self-defeating. As more and more enthusiastic citizens start asking questions, the planning machinery begins to get bogged down. Elapsed time between questions and answers keeps growing, and public enthusiasm declines. The result is interested but frustrated private citizens who feel they cannot get answers and well-meaning but frustrated transportation planners who would like to answer the questions but cannot get to them for two months.

The logical solution to the dilemma of increasing responsibility and decreasing resources is increased productivity. Michigan has a unique opportunity in this respect. Michigan's Statewide Transportation Modeling System (STMS) is an operational multimodal impact-analysis system that has been used in more than 350 applications over the past two and one-half years. This system has allowed MDOT to take full advantage of interactive graphics through the development of the NETEDIT computer program. NETEDIT is an interactive method of creating, displaying, and analyzing alternate transportation networks and of relating associated socioeconomic data to them. Through interactive graphics, NETEDIT has (a) decreased the workload, (b) decreased elapsed time, and (c) increased accuracy, thereby allowing MDOT to plan, design, and construct a transportation system more efficiently (1-4).

HARDWARE REQUIREMENTS

There are only two basic hardware requirements for implementing an interactive system of network display and updating. First, the user needs a cathode-ray-tube (CRT) computer terminal (Figure 1), which is similar in appearance to a television screen and is capable of drawing pictures as well as printing alphanumeric characters. Second, there must be access to a computer that supports a high-level programming language and allows a computer program to be linked with routines that actually do the drawing on the screen. These routines can usually be purchased from the manufacturer of the CRT at a nominal cost.

It is important that the screen of the CRT be large enough to display a natural unit of data--say, a county--without appearing cluttered. If it is difficult for the operator to figure out what he or she is looking at, the objectives of increased efficiency and accuracy will not be realized. The most popular screen sizes range from a rectangle about 10 in measured diagonally to one that is about 25 in diagonally. The examples in this report were made on a Tektronix 4014-1, which has a 19-in screen.

Resolution is important if the program is to calculate the actual length of a link from the screen coordinates of its end points. For example, the 4014 has 4096 x 4096 addressable points. By comparison, the small-screen 4010 has only 1024 x 1024 addressable points, which somewhat increases the chance of approximation error. The user may decide approximately how many square miles should be displayed on the screen and then compute possible errors for each model of CRT being considered.

A third consideration is computer speed. Most high-level machines are quite fast; however, if a time-sharing system is overloaded, one should think very carefully about investing the time and money required to put up an interactive graphics system. Moreover, even a fast line will not help in such a situation: What good does it do to draw at 9600 baud if there is a 30-s pause between bursts of drawing?

Finally, the reliability of the time-sharing system to be used must be looked at objectively. If the system is prone to frequent periods of down time, this will produce many false starts and restarts in a network update. In that case, it would probably be less frustrating and more productive to stay with manual coding techniques.

NETEDIT is now running (with excellent results) on a large-scale Burroughs B-7700 computer with three processors and a 4.7-megabyte memory. An earlier version was run on a Control Data 6400-series machine. It is written in FORTRAN-IV and uses Tektronix Plot-10 graphics subroutines. The program is currently limited to networks of not more than 13 000 links and 16 000 nodes.

STMS NETWORK CONCEPTS

Collection, storage, and retrieval of all data employed within STMS are tied to a 547-zone system. Of this total, 508 are in-state zones. The zonal concept is of extreme importance in that it provides a dynamic link between information retrieval and actual model procedures. The conversion of raw data from storage within the information files into accurate travel, social, economic, and environmental indicators has been effectively accomplished as a result of gearing the entire system to the zonal format.

The transportation network model is the means by which the transportation planner describes to the computer in its own language the transportation system under study. The network is defined in the network model as a set of links and nodes. Nodes are numbered points located by X-coordinates and Y-coordinates that reference a statewide coordinate grid. A link is defined by its connection of the nodes of two networks. Figure 2 shows a conceptual drawing of a portion of the highway network within several zones of a zone system. The illustration indicates that there are two basic types of system links--regular links and pseudolinks, which are known as centroid links. A regular link is used to describe a section of the transportation facility (e.g., highway or rail segment), whereas a centroid, in connecting itself to a node of the base network, allows the feeding of traffic to and from a zone and off and onto the system.

The transportation planner must differentiate between types of links according to certain physical and travel characteristics. Each discrete piece of link-specific descriptive data, or link attribute, is stored on a computer file in what is known as a volume field. An understanding of the volume-field concept is critical to one's comprehension of the

Figure 2. Network description.

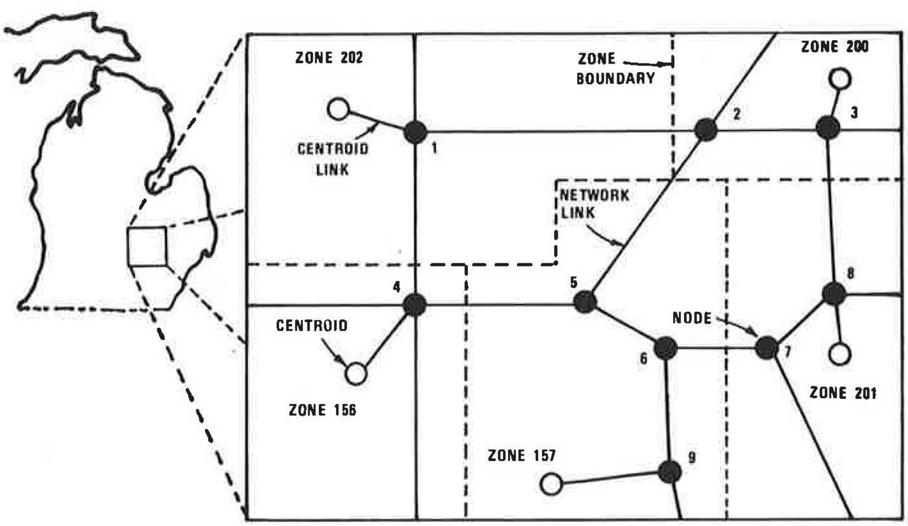
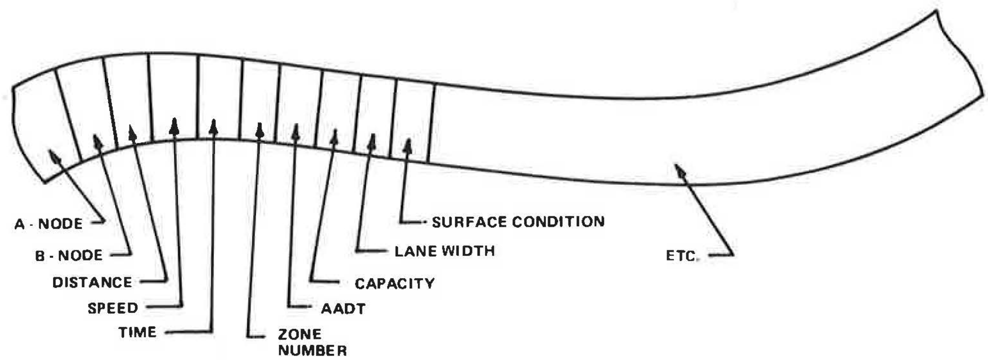


Figure 3. Link description.



modeling system, for it is consistently employed throughout the process. Figure 3 illustrates how a highway link's attributes might appear on a segment of magnetic tape if it were visible to the human eye. In creating the network model, data records are taken from punched cards and recorded sequentially on the network data file. First, a link's A- and B-nodes are recorded to distinguish it from other links within the network. This initial portion of a link's data file contains, in volume fields, other information pertinent to its description, e.g., type (existing or newly created) and jurisdiction (who funded the construction and maintains the facility).

NETWORK UPDATING: MANUAL VERSUS INTERACTIVE

Once the network file is created through standard coding procedures and specialized computer programs, it may be modified to simulate alternative transportation proposals. Many such changes are performed when comparisons of travel impacts are desired. The transportation network model provides a primary input not only for the travel forecasting model, but for all models developed within the statewide system. The steps involved in creating and modifying networks are exacting and time consuming. These are the steps that the interactive network updating system was developed to replace:

Step 1: Cards must be punched to update a base network into the alternative network. For the analyst, this means coding (i.e., writing on forms from which cards will be punched) all the links and nodes to be added, deleted, or changed in the updated alternative network and the link attributes for the new or modified links.

Step 2: After the cards have been keypunched, they are input into a series of computer programs that (a) alter the base network to produce an updated alternative network, (b) change or add volume-field (link-attribute) information, and (c) create a computer-readable version of the network that can be plotted on paper.

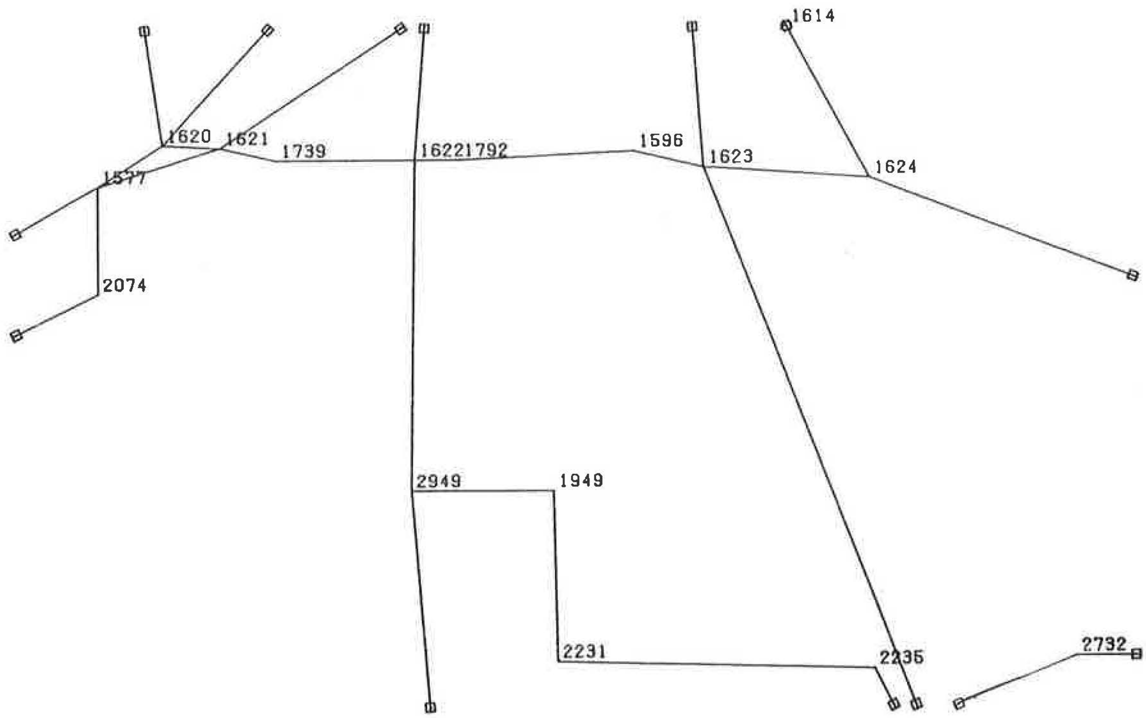
Step 3: After the network has been plotted, which produces a plot such as that in Figure 4, the analyst pores over it to make sure that no errors have been made. Typically, the analyst must check (a) the location of modified nodes, (b) the correct addition or deletion of all links, and (c) the correct volume-field updating. If errors are discovered, further updating is begun by going back to step 1.

Often the entire process is repeated two or three times. Even if the procedure is gone through only once, it usually takes up to two weeks for coding, computer runs, plotting, and plot checking.

In comparison, the statewide interactive network updating system is simpler and less laborious and provides for immediate verification. From the user's perspective, the system can be a simple one-input one-output system such as that shown in Figure 5. The user merely specifies the necessary information to access the base network at the beginning of the program; all other file manipulation occurs internally. The user need only pay attention to the location of roads when the desired network alternative is being built.

NETEDIT currently manipulates networks stored in

Figure 4. Plot made by using Cal Comp network plotter.



Burroughs' TP-System packed form. However, the program is modular enough and its concepts are general enough so that modifying the packing and unpacking subroutines should be all that is necessary to allow it to read and write networks in other forms, such as DCO/TRANPLAN or the Urban Transportation Planning System (UTPS). After the updated network has been written to a disk or tape file, it is used like any network created by batch processes. Thus, NETEDIT uses no special interfacing mechanisms other than its own packing and unpacking routines. Internally, the program creates and manipulates several background files (Figure 6):

1. Link attributes--capacity, base-year counts, and up to 45 other pieces of link-specific data--are stored on a randomly accessed disk file to save core space.
2. At any point between unpacking and packing, the current intermediate state of the network may be written to a disk file by giving the command SAVE. That stored state may be restored at any time by the command RCVR (recover), which provides a check-point and restart capability.
3. As the user executes network modification commands, they are encoded and written to a "tank" on a disk. After any system failure short of a disk crash, the user need only unpack the network that was being worked on when the system went down and execute the command TANK. Then the network will be changed as if every command had been reentered.
4. A record of changes made by every user to every network is kept in a log file on a disk pack. The file is periodically printed out and rewound. Besides providing informational backup, the log is also handy for answering questions about why the program does not work correctly.

SYSTEM DEMONSTRATION

There are five primary application categories for which NETEDIT is typically used on a daily basis: (a) network updating, (b) network display and presentation, (c) network generation, (d) tree (path) plotting, and (e) socioeconomic data display. By considering each category in turn, the reader may see the extent and versatility of the NETEDIT command set. Commands may usually be given in any order, so that the user may base the next action on the results of the last. This characteristic also makes NETEDIT a useful tool in dealing with private citizens' inquiries, in which the answer to one question often prompts another. Figures 8-19 were created by a hard-copy unit similar to the one shown in Figure 1. More than 9000 such copies have been used in day-to-day planning over the past two and one-half years in Michigan.

Updating

Consider now the portion of the highway net around Grand Rapids shown in Figure 4. Suppose MDOT is asked to evaluate the impacts of putting in a bypass around Grand Rapids (Figure 7). The planner would begin by adding a link (ADDL) from I-196 to US-131 (nodes 2074-2949), as shown in Figure 8. In Figure 9, the change-parameter command CHGP is used to reconstruct county road 2949-1949 to a four-lane freeway. The command CJ, 3 changes the road to jurisdiction 3 and puts in all volume-field defaults for that jurisdiction; then the command 3, 12 changes parameter 3 (link type) to 12, and the command 5, 1 changes parameter 5 (freeway code) to 1. The command DISP redraws the picture.

In Figure 10, an interchange is inserted on 1623-1961 at node 2231 by using the split command SPLT. This is equivalent to adding node 2231, deleting link 1623-1961, and adding links 1623-2231 and 2231-1961; during interactive operation, 1623-1961 is Xed out, which shows the deletion. The two new links are given all the characteristics of the parent, and the sum of their distances is made equivalent to that of 1623-1961. Finally, links 1949-2231 and 2231-1624 are added to complete the bypass.

This small example points out an additional ad-

Figure 5. Network updating from the user's perspective.

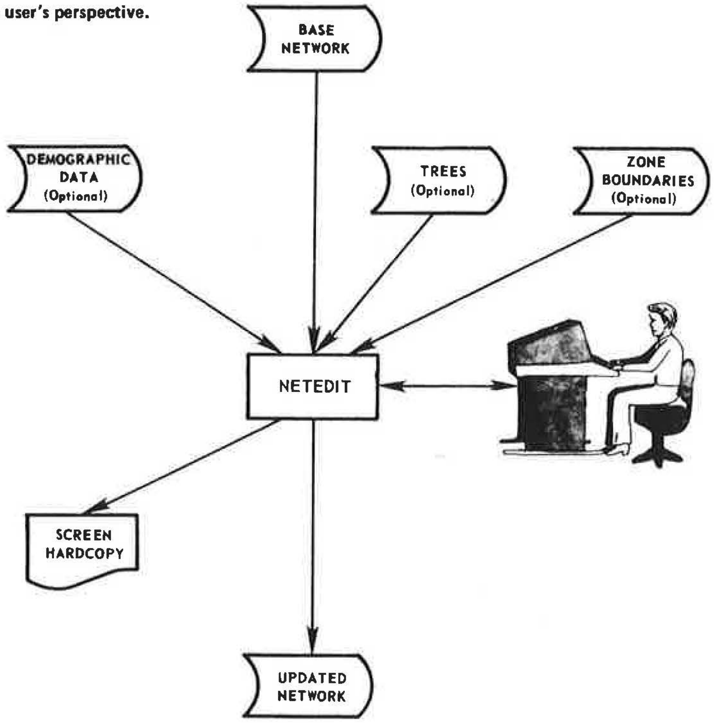
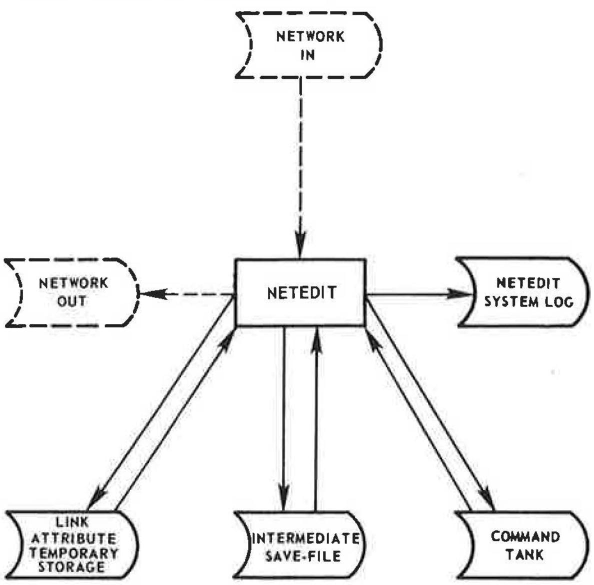


Figure 6. Program background files.



vantage to interactive updating--instant feedback. The user knows immediately whether the correct link has been deleted or whether a link to be added was placed correctly.

Display

Whether the planner is investigating a portion of a network in detail, finalizing dimensions and parameters in preparation for an ink plot, or merely producing quick 8.5x11-in plots for a meeting, NETEDIT cuts elapsed plot time from hours or days to seconds. For example, it is possible for the program to vary the manner in which a link is drawn based on link attributes. In Figure 11, the solid lines are state trunk lines, dashed lines are county roads, crosshatches are rail links, and links marked by triangles are rail-to-highway connectors. One may also choose, or filter, links into or out of the display based on attributes. Figure 12 is the result of filtering out all but rail links from Figure 11. A portion of the current display may be enlarged for further study: Figure 13 is the result of selectively windowing in on the right half of Figure 11. Zone boundaries can be superimposed by using dot-dash lines (Figure 14). Link attributes can be displayed either numerically (Figure 15) or by bandwidth (Figure 16--attribute 7 is 1975 average daily traffic).

Network Generation

A digitizer tablet attached to a graphics terminal (such as the one shown in Figure 1) can be used to great advantage both in the initial creation of a network and in the addition of an existing network. The user simply tapes a map to the tablet, enters two fixed points by using the bull's-eye cursor or stylus, and informs the program where these points lie in the master coordinate system. From then on, one-letter cursor commands control the flow of the subroutines as the user moves with the cursor from node to node, automatically creating links between nodes. In this manner, it has been possible for persons who have only moderate familiarity with computers in general--and NETEDIT in particular--to digitize detailed networks of approximately 2000 nodes and 2000-3000 links in two weeks. Because the tablet routines run as a subset of NETEDIT, one need not jump back and forth between programs to edit links entered from the tablet and, because the final product is a packed net, no further batch processing is necessary. This capability now allows Michigan to digitize an A-node, B-node network for all roads in the state. This network will be the basis for future impact analysis and needs studies.

Tree Plotting

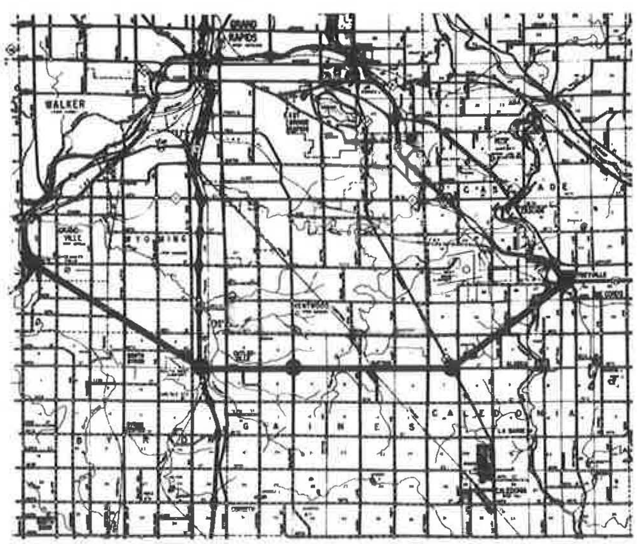
In network calibration as well as in alternative analysis, a large part of the job involves verifying that driving paths on the model reflect reality. Usually this is done by a batch-driven off-line plotter package. However, because most agencies have only one or two plotters to serve all users, a bottleneck often results.

The use of interactive graphics to plot trees offers several advantages, not the least of which is the reduction of turnaround time from days to minutes. In fact, interactive tree plotting has proved to eliminate as much as 90 percent of the time elapsed in highway network calibration.

Also, because the tree plotter is a subroutine of NETEDIT, all the options available in editing or displaying networks are accessible--windowing, changing line type based on link characteristics, and link-attribute annotation. One could even plot trees by bandwidth, the width of the line being determined by, say, accident rate.

Figure 17 shows a portion of a multimodal tree from Michigan's STMS. Note that in the interchange from zone 33 to zone 404, the tree proceeds on state trunk line to the rail-to-highway connector, continues onto the rail line, leaves the rail line, and finishes its path on trunk line.

Figure 7. Proposed Grand Rapids south bypass.



Socioeconomic Data Display

Any data that can be associated with a node on a network can be superimposed on a network plot. Figures 18 and 19 show the same data--projected 1980 zone population--plotted as vertical bars and as circles; the height of the bar or size of the circle indicates the magnitude of the zone's population relative to the other zones in the state. In these examples, zone boundaries have been drawn as dot-dash lines by NETEDIT. This subroutine becomes useful in attempting to relate a particular travel impact to the socioeconomic classes it affects.

These examples have demonstrated only a few of the NETEDIT commands. A more comprehensive list follows.

Network Updating

Command	Definition
ADDL	Add new link
ADDN	Add new node
CHGP	Change link parameters
DELL	Delete link
DELN	Delete node and any link that uses it
MOVN	Move node
SPLT	Split link in two; add new node

Network Display

Command	Definition
DISP	Display network in current coordinate window
ENLG	Enlarge portion of network
GOTO	Center window around specified nodes
OLVW	Return to previous virtual window
ZONE	Superimpose zone boundaries
NOZN	Turn off zone boundary overlay
NODE	Display node numbers
CENT	Display centroid numbers only
BLNK	Turn off node annotation
PAGE	Move window one screen page in one of eight directions
PARA	Display link attributes
FLTR	Set filter criteria for inclusion of link in display
CHGF	Change existing filter criteria
LINE	Change line type for link type or jurisdiction group
BSET	Set key attribute and band limits for bandwidth plotting
LGND	Display legend for bandwidth plot

Figure 8. Network update: adding a link.

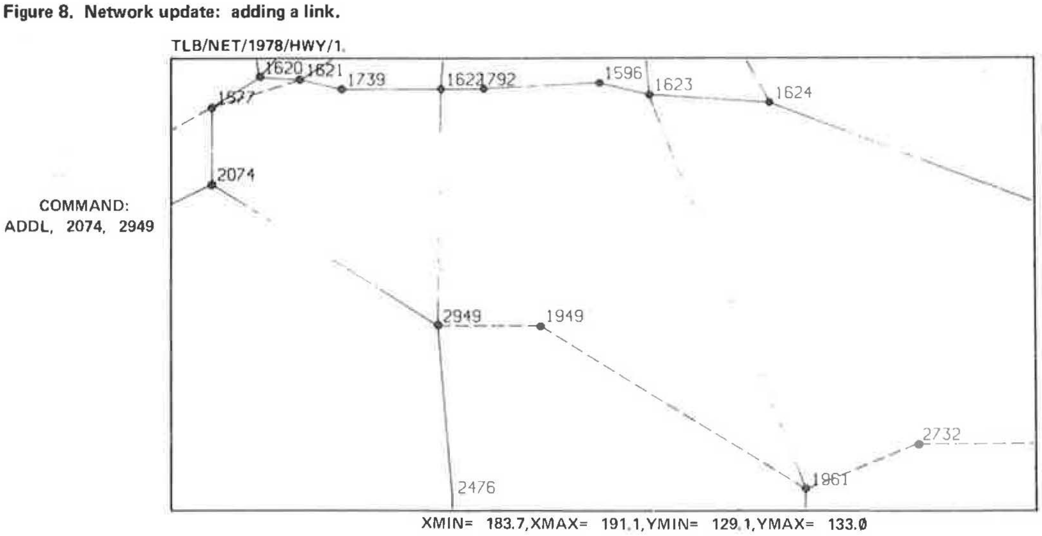


Figure 9. Network update: reconstructing county road to freeway standards.

COMMAND:
CHGP, 2949, 1949
CJ, 3
3, 12
5, 1

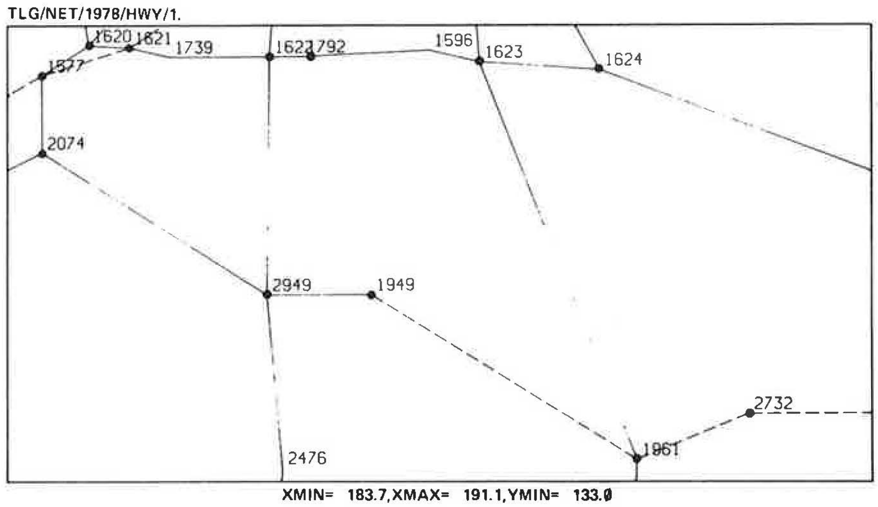


Figure 10. Network update: adding an interchange.

COMMAND:
SPLT, 1623, 1961
ADDL, 1949, 2231
ADDL, 2231, 1624

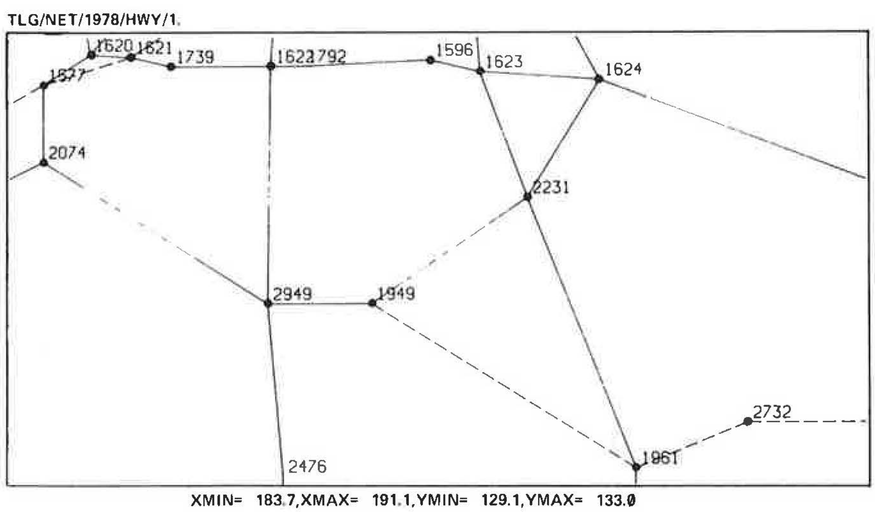


Figure 11. Line types based on link attributes.

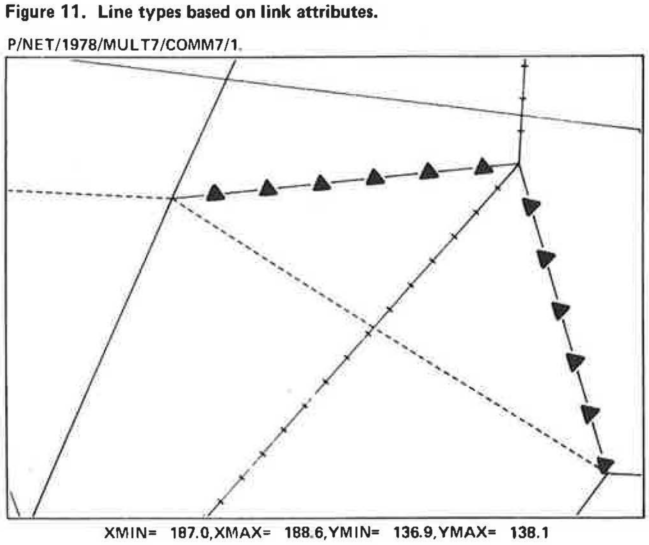


Figure 12. Filtering.

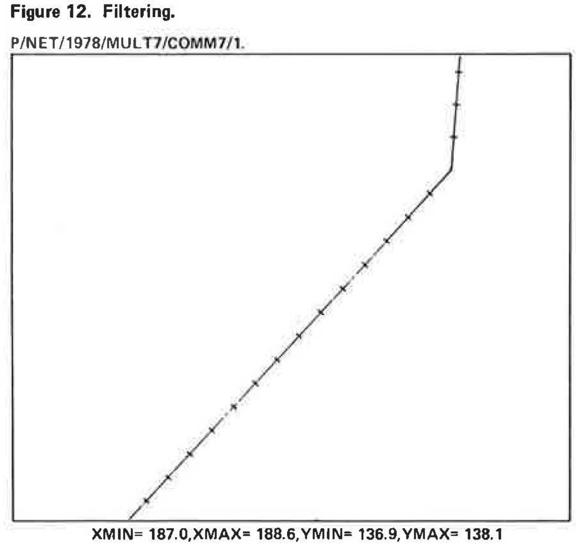


Figure 13. Windowing.

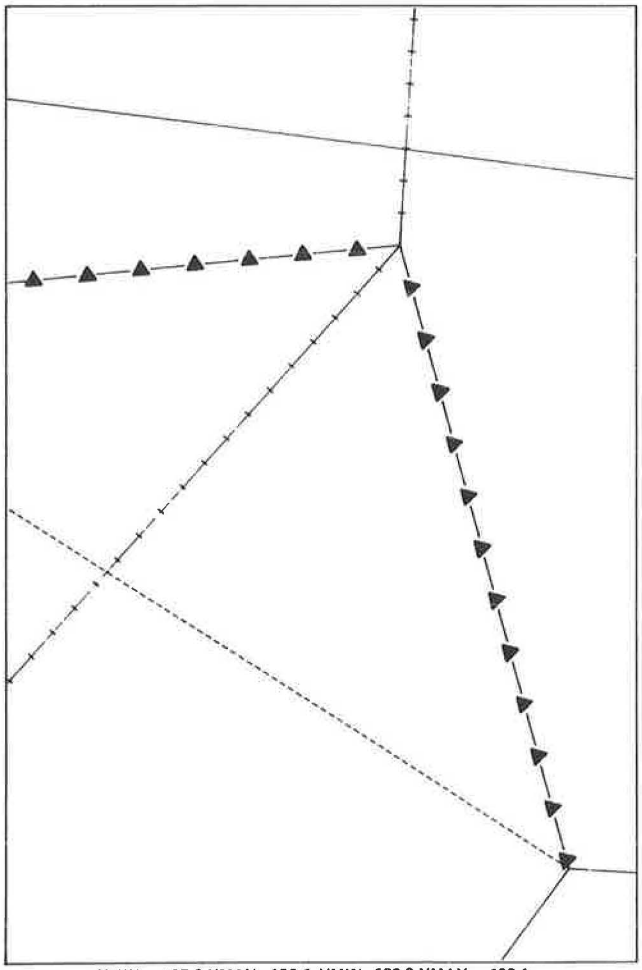


Figure 14. Zone boundary overlay.

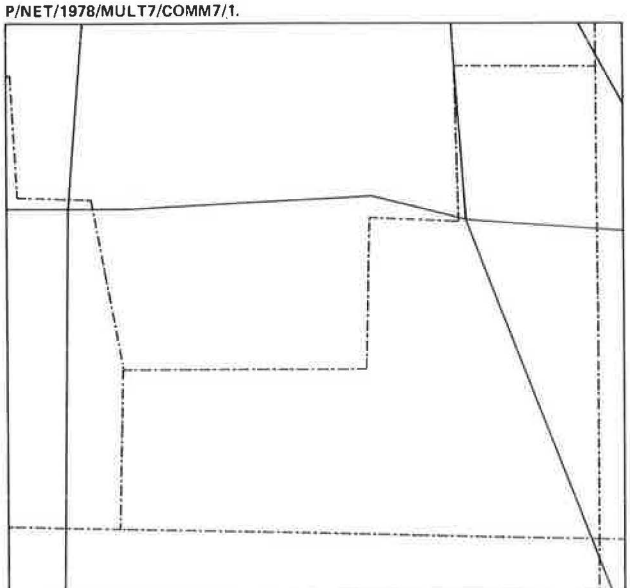


Figure 15. Numeric display of link attributes.

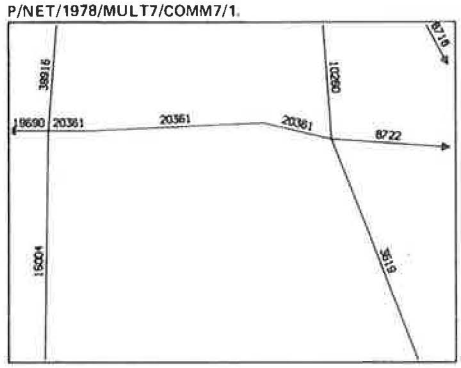


Figure 16. Bandwidth plotting.

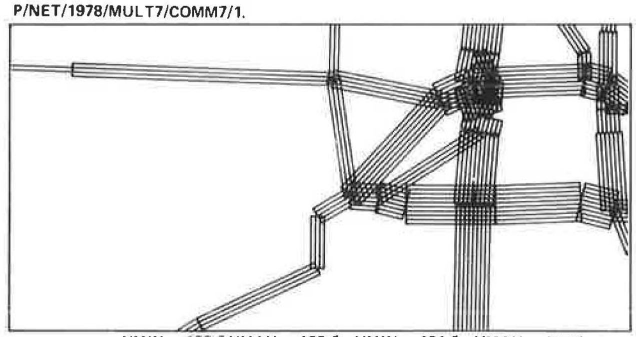


Figure 17. Multimodal tree plotting.

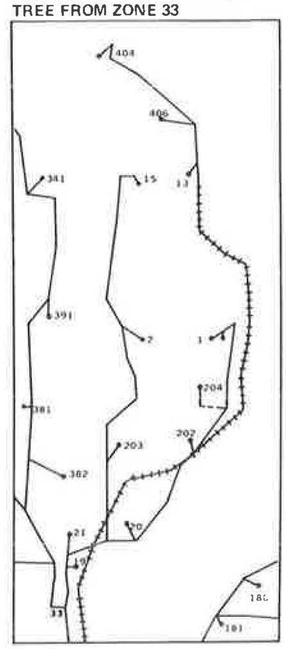


Figure 18. Socioeconomic data displayed as vertical bars.

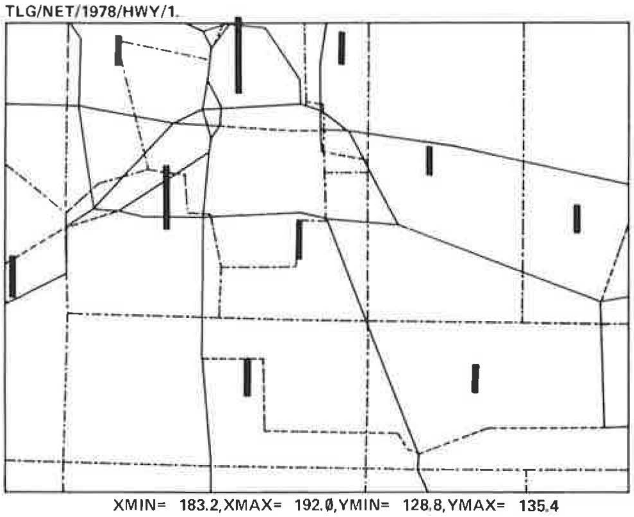
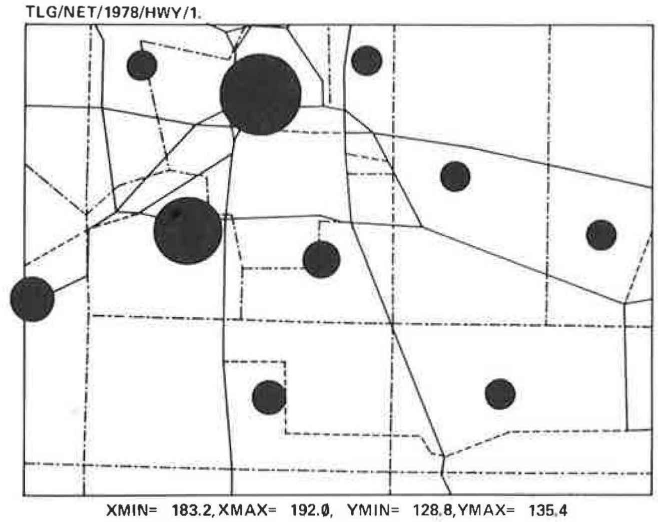


Figure 19. Socioeconomic data displayed as circles.



In addition to all regular commands, for network creation the following cursor commands may be entered while the graphics tablet is receiving points:

Network Display

Command	Definition
A	Automatically number new nodes
D	Delete links or nodes
E	Enlarge
F	Window full county or region
M	Move node
N	Return node sequence to existing node
O	Return to previous virtual window
P	Plot; redraw picture
R	Return to main program
S	Split link
#	Same as NODE
B	Same as BLNK

Tree Plotting

Command	Definition
TREE	Read tree file and plot portion of tree in current window
NEWT	Close old tree file, open new one, and plot tree
TIME	Annotate cumulative times on links of tree

Socioeconomic-Demographic Data Plotting

Command	Definition
DEMR	Read node data from disk
DSET	Establish level cutoffs for node data
DEMB	Plot demographic data as vertical bars
DEMC	Plot demographic data as concentric circles

Miscellaneous Commands

Command	Definition
FIND	Print out X-coordinates and Y-coordinates for up to four nodes
SAVE	Save current network configuration in temporary form on disk
RCVR	Return to last SAVE
TANK	Recover command tank and execute encoded commands
CHGH	Change information in network header file
PACK	Create output network in bit-packed form

CONCLUSIONS

In a time of increased emphasis on public involvement in the transportation planning process, it is important to ensure that the process of generating alternatives remains responsive to public feedback. Although the number of alternative plans to be considered increases dramatically, the planning system cannot allow itself to get bogged down. Time elapsed between citizen question and system answer must not become unreasonably long; this would have the ultimate result of killing the public's newly found enthusiasm.

The interactive network graphics program described in this paper has helped relieve a potentially crippling burden on Michigan's planning procedure. It has eliminated a major part of the cost and time elapsed in generation and evaluation of alternatives. Because NETEDIT is much more interesting to use than manual methods, it also tends to eliminate a majority of the errors that can occur after hours of network coding.

Finally, it must be stressed that NETEDIT is a production technique used in hundreds of statewide model applications. It is the proven productivity that makes NETEDIT such a valuable aid in transportation planning.

REFERENCES

1. T. L. Gotts and L. G. Scott. Michigan's Statewide Transportation Modeling System, Volume 2-A: Efficient Network Updating with Interactive Graphics. Michigan Department of Transportation, Lansing, May 1976.
2. T. L. Gotts. Michigan's Statewide Transportation Modeling System, Volume 2-A, Appendix A: NETEDIT Command Directory. Michigan Department of Transportation, Lansing, June 1977.
3. T. L. Gotts. Michigan's Statewide Transportation Modeling System, Volume 2-B: Tree Plotting with Interactive Graphics. Michigan Department of Transportation, Lansing, Jan. 1977.
4. J. Newell. Michigan's Statewide Transportation Modeling System, Volume 3-B: Automatic Network Generator Using Interactive Graphics. Michigan Department of Transportation, Lansing, Aug. 1977.