Possible Futures for Traffic Simulation

Paul Ross

This paper discusses the ultimate future of traffic simulation--that is, what features will traffic simulation programs have 20, 30, or even 50 years from now? The statements made here are necessarily hypothetical and subjective. No apology is made for that; there just does not seem to be any other way to cover the subject.

In this discussion of the future of traffic simulation, I will exclude ideas that are currently under development or planned. (See the paper by Radelat elsewhere in this proceedings.) This is not to say that no research is under way on these subjects. Indeed, I am aware of pilot studies or preliminary research on nearly all the features that will be described here. Nevertheless, it does not appear that many of these features will be incorporated into publicly released traffic simulation programs within the next five years at least.

TRENDS IN COMPUTER TECHNOLOGY

There are many things that will affect the future of traffic simulation. The most important of these will be the future of traffic itself. What kind of vehicular traffic will exist in the year 2000? smaller vehicles? larger vehicles? Will there even be individual vehicles? There certainly are conflicting trends in vehicular traffic at present, but it is far beyond the scope of this paper to sort them out. For the purposes of this paper, let us assume that traffic will not be radically different from a collection of individual vehicles as it is now. Let us further assume that no other changes--except perhaps in computers and computer technology--need to be predicted at this time. Such a restriction is necessary to limit the scope of the problem; otherwise, it would be necessary to deal with an impossibly broad topic.

What, then, are the likely changes in computers? And how will these changes affect traffic simulation? The following is envisaged.

Mainframe computers will get bigger, faster, and more expensive. This is a simple extrapolation of a well-known current trend. The price per calculation will continue to go down, although not as rapidly as it has in the recent past. Since manpower costs are essentially negligible in computer calculations, the cost of an individual calculation will remain insensitive to labor costs and will not rise in terms of real dollars. However, since the capital cost of individual mainframe computers will increase as their computing power increases, we will see fewer and fewer organizations able to afford the most powerful computers.

Small computers will become more powerful, and powerful computers will become smaller. We already have computers with full abilities that are small enough to carry in a briefcase (although they are hardly pocket-sized yet). The usual office will replace its typewriters with word processors that will be cheaper than typewriters--very much in the way that pocket calculators have driven the old mechanical desk calculators out of the market. These word processors will have computational capabilities fully able to run traffic simulation programs.

These devices might be better thought of as computers that also do word processing but they are more likely to be justified on the budgets as office equipment than as laboratory equipment. The more-deluxe versions of the word processors on the market now already have the ability to do mathematical calculations, and soon the cost of adding programmable scientific calculations will be just a few dollars--the cost of a silicon chip. Soon we will be able to run NETSIM on our office typewriters.

Public time-sharing services will excel at providing service to small or medium-sized organizations whose computing requirements fluctuate widely. One can easily visualize a small consulting company that needs negligible computing time except that once a week it runs a large simulation that requires an hour or two of CPU time and a few megawords of random-access storage. Clearly it will not be cost-effective for such an organization to purchase its own computer and it will have to turn to a public computer-sharing service. It will be in the interest of such a service to use the most powerful available computer in order to service as many customers as possible simultaneously. Extraordinary demands will probably be satisfied on a batch-mode basis.

Organizations with reasonably constant need for computing power will tend to buy their own in-house computer since they will be able to choose one to match almost exactly their individual requirements. These organizations will probably provide real-time operation on small computers or time sharing among several individuals on medium-sized computers simply because human time will be worth more than computer time. Batch-mode processing at night or on the weekend may be required for large jobs within such a setup.

Every computer powerful enough to run a traffic simulation program will have some form of graphic output. As a matter of fact, the graphical devices will be cheaper than hard-copy printers. Liquid-crystal matrix displays can be made without all the complicated moving parts that are inherent in hard-copy printers. When cathode-ray tube or liquid-crystal displays become common, there will be no reason to restrict the outputs to alphanumeric characters and most output displays will have full graphic capabilities. So, eventually, the office typewriter will not only be able to compute, it will also be able to produce pictures.

FUTURE DEVELOPMENTS

How will the above trends in computer usage affect traffic simulation? The following are speculations about future developments. They are arranged in order with the most certain and immediate prospects first and the most speculative and remote ideas last. Indeed, experimental versions of the first three ideas are already in use; it is just that no development or release of a traffic simulation program with these features is scheduled at the present time.

Graphic Displays

The surest thing is that simulation programs will make greater use of graphic displays. Since virtually all computer terminals will have a graphic device as its normal form of output, this development is inevitable. [The only thing that has held up the incorporation of graphics into NETSIM has been the fact that there is no common graphic language. The language used for computer graphics depends on the make of the terminal; it is not like FORTRAN, which can be executed on virtually any brand of computer.]

At first, NETSIM outputs will show such things as the queue lengths at all the intersections and various other forms of output information. Such displays will allow the user to grasp the overall operation of a network at a glance, which will be much quicker and more meaningful than wading through the reams of computer printout that are now presented.

Graphical output will be followed by graphical displays of the simulation program in operation. Pictures of little cars running around the network are generally thought to be a good public relations tool. That is, they are the kind of thing one likes to show when explaining one's results to a somewhat dubious committee of nonexperts. Animated operating displays are certainly useful for such explanations, but they are even more useful to the practitioners themselves. There is no more certain way to find mistakes in the input than to look at how the computer thinks the system is supposed to operate. Left turns coded as right turns or obvious mistakes in signal phasing stand out immediately. The technology to show full animation has been available for only a few years and is currently very expensive. However, it is certain to become cheaper and, over the time span we are considering, should become readily available.

Interactive Calculation

Traffic simulation programs will usually be interactive. That is, the operators will be able to interrupt the programs during execution and change various parameters. This interactive ability will be a natural outgrowth of the use of graphic terminals. Widespread use of graphics will, by itself, lead to more interactive programs. While it is possible to run a program in batch mode and then look at the outputs generated by the computer sometime later, this is not a convenient or natural way to use computer graphics. With graphic displays it is natural to have the computer instruct the display device to draw some complicated picture and then await confirmation that the picture was indeed drawn before proceeding with the next calculation. Consequently, it is a very small step to allow the operator to interrupt and change the program since the computer is waiting for a response from the terminal anyway.

With a time-sharing option, or dedicated operation on a small computer, at least some small amount of interactive computing seems inescapable. At a minimum, the program will analyze the input data and inform the operator of obvious errors before he or she leaves the terminal. A simple program could operate this way but we will soon see programs that ask the operator for input data in plain English and analyze it item-by-item for obvious errors and consistency with previous data. The operator will be informed of problems before his or her attention has moved on to the next data item.

Until graphic devices become common, this may be all the interactive capability that will be useful. But once the operator can see how the entire network is operating at a glance from some animated operating display, he or she will want to be able to control that operation. Adding interactive abilities during program execution will be natural.

On-Line Simulation

The interactive and graphic display features will lead to "on-line simulation" for traffic control systems. On-line simulation refers to a service provided to operators of computer traffic control systems. With this feature the operator, at the touch of a button, will start an interactive graphic simulation running. The program will start with initial conditions that are identical to those that are current in the real network at the time the button is pushed. If the program runs four or five times as fast as real time, the operator will be able to foresee events in his or her actual network and possibly test alternative strategies.

There are a myriad of cases where such ability would be useful. One example is a situation in which an accident completely closes a network link. Even if the control algorithm is able to provide an appropriate response to such a traffic situation, it will be useful to foresee how the traffic disturbances will propagate so that police can be dispatched appropriately and, perhaps, news media notified of impending congestion at critical locations.

The ultimate stage in on-line simulation will be a program that runs continuously and checks itself against the real traffic. In this way, the simulation program can adjust itself to changes in the vehicle mix and driver behavior without any human intervention.

Data Acquisition

As users of NETSIM and other microscopic simulation programs know, input preparation and data collection are inordinately tedious and expensive. There is a very real need for "automatic input" to such simulation programs. Automatic input here means providing accurate geometric data (such as link lengths, grades, and corner radii) and traffic data (volumes, turning movements, and traffic composition) with little or no human intervention.

For a start, it is suggested that aerial photographs projected onto a digitizing tablet would be quite useful. Link geometry could quite accurately be entered just by touching origin and destination nodes. Corner radii could be entered if needed. A single aerial photograph is not much use in estimating volumes, but the simulation programs could be easily written to use density (vehicle/mile) instead. Input that starts from cars at specific locations throughout the network would have the additional advantage that no initialization period would be needed before the simulation results are valid. A great majority of the input now required for the NETSIM program could be entered just by touching points on a digitizing tablet. While this would require substantially different forms of data input processing, the basic principles of NETSIM operation would not be affected. The technology to do all this is available now.

This procedure might correctly be termed semiautomatic input because a human operator must participate by pointing out the nodes, cars, trucks, corner radii, etc., to the computer. Is there a possibility of more-nearly-true automatic input? Yes, Sensor for Control of Arterials and Networks (SCAN) technology could be adapted to a completely automatic input system (1). SCAN is a television-based detector system in which the computer identifies the images of moving vehicles and tracks them over space and time. This technology could be adapted so that aerial motion pictures could be analyzed and virtually all the simulation input could be assimilated into the computer without human intervention. The SCAN detector could pick up the network geometry, volumes, and turning movements automatically. In effect, all we would have to do would be show the computer a movie of the network operation and the computer would be able to simulate it.

Integrated Simulation

Finally, all these features will be integrated to produce full, citywide simulations. Such programs will be fully microscopic (as NETSIM now is). We will show the computer an aerial motion picture and the computer will identify all the fixed and the moving objects. It will be able to classify the moving objects automatically as small, medium, or large automobiles; trucks; transit buses; school buses; fixed-rail transit; etc. It will even, if so ordered, identify all the pedestrians. After identifying these objects, it will deduce the origin-destination table for each class of moving object. The program will deduce the acceleration-deceleration curves for each class of object and the distribution of headways for "followers."

In short, the computer program will be able to assimilate all the information that it needs to run a complete simulation of everything that moves in a whole city--all the statistical distributions and all the adjustable parameters. It will measure these quantities and not merely assume characteristics measured in some other city.

The graphic output from this citywide simulation will be extraordinarily lifelike. Computer techniques already exist to identify and manipulate elements of pictures while maintaining photographic realism. The simulation program will identify the fixed-background photograph and maintain it continuously on the output display. The simulation program will also have identified which photo elements represent cars, buses, trucks, trains, pedestrians, dogs, cats, etc. The output will have these photo elements superimposed on the fixed-background picture and moving in lifelike ways. When the simulation program generates a new vehicle it will represent it in the output with a photo element chosen at random from those photo elements that were identified as being members of the same vehicle class.

CONCLUSION

None of this is particularly visionary. The techniques to accomplish all of these things already exist--although in cumbersome and expensive experimental forms at present. It is not a question of "Can these things happen?" They can. There is no doubt about that.

Will they happen? Yes, probably. As long as there are research programs.

Will any of these techniques become common? That is pure guesswork. It depends on so many things: government regulation, economic climate, public concern. We will not discuss how common these techniques will become; this paper has been speculative enough without going into those matters.

However, we can briefly speculate about items that will not become popular, although they are known to be feasible. For instance, a traffic simulation language will probably never become popular. General simulation languages such as SIMSCRIPT and Q-GERT serve a real need for persons who have to simulate unique operations, but there are not enough persons working in traffic analysis to support a comprehensive traffic simulation language. Moreover, traffic situations do not vary so much that they cannot be all encompassed in a single program. It is hard to define the dividing line between what is a "program" and what is a "language". A very general and flexible program could be regarded as a language by some persons. Certainly traffic simulations will become more general and flexible but the effort to keep them easy to use will maintain their identity as programs, not languages.

Finally, I believe that the best simulation programs will not incorporate optimization. Of course, the optimization programs that are already in use do incorporate some form of simulation or evaluation. However, they are not the most accurate forms of evaluation and there are reasons why they cannot be.

The most efficient forms of mathematical programming, such as linear programming, require that the system model have certain mathematical simplifications. (Linear programming, for example, requires that the model must be piecewise linear and the region of feasible solutions must be convex. Other techniques require other restrictions.) The optimization methods that use only the model output and make no assumptions about the form of the model (hill-climbing or other gradient methods) are inherently inefficient and cannot guarantee a global optimum. It is beyond the scope of this paper to argue why the best simulation models will always be microscopic in character, but it is obvious that a microscopic simulation program cannot be used with an efficient and accurate optimization technique. Therefore, accurate and efficient traffic system optimizations are inherently impossible and there is no point in even trying to use the best evaluation models. Equal accuracy can be achieved by using simple, but good, models and quick and accurate optimizations. On the other hand, it is likely that in the future all signal optimizations will be done on-line.

ACKNOWLEDGMENT

The ideas presented here are mine and do not represent the opinion of FHWA. In particular, the reader is cautioned not to draw any conclusions as to lines of research that FHWA will or will not undertake based on this discussion.

I am indebted to George Tiller, Guido Radelat, and many others whose ideas have been used without acknowledgment.

REFERENCE

1. Sensor for Control of Arterials and Networks (SCAN). Breadboard Hardware; FHWA, FHWA/RD-80/024, Oct. 1980.