Review of Road Traffic Network Simulation Models

Paul Rosa, Office of Research, and David Gibson, Office of Development, Federal Highway Administration

Proposals for computer traffic simulation date from 1951, and the first actual documented simulation was performed in 1955 on an analog computer. Since that time, many simulation models have been developed as knowledge of traffic principles has increased and computer facilities have improved. Such traffic simulation models may be placed in three groups: single road, single intersection, and traffic network. This paper discusses only the last. Nineteen network simulation models are discussed. These are grouped into ten obsolete models, six traffic network models suitable for current computers, and the simulation portions of three signal optimization programs. A brief description of the operating principles and unique features of each model is given, and the level of modeling detail, model flexibility, and usefulness of the output are assessed. Validation efforts on the model are considered, and, where known, computer language, type of machine, core requirements, and speed of execution are given.

There are three classes of traffic simulation models: single road, single intersection, and network. Hsu and Munjal (2) have reviewed the single road freeway models, but rural road models, which reproduce the effects of sight distance restrictions, rolling terrain, and passing opportunities, are rare. Single intersection models have generally been built for a specific purpose and are not usually available as a consumer product. This paper discusses the currently available network simulation models and some now obsolete network models. We also summarize three signal optimization programs that include traffic network simulation.

Modern traffic simulation programs use digital computers and time-oriented bookkeeping; the event-oriented bookkeeping associated with most simulation languages seems inappropriate. Some models (macroscopic) treat the traffic stream as a continuum and conceptualize the traffic stream as a fluid. Individual vehicles are not identified in macroscopic models; indeed, a network element may even contain fractional vehicles. Macroscopic models generally are very economical of computer storage and very fast in execution, because they do not have to keep track of each vehicle. However, they cannot represent the traffic stream in the detail that many traffic engineers want.

Platoon models are a half-step toward detailed realism. In these models, vehicles are grouped into platoons, and the program keeps track of the location, speed, acceleration, and size of the platoon. Platoon speed, usually a function only of the general density of vehicles in the vicinity of the platoon, avoids complicated car-following calculations.

Microscopic simulation models are the ultimate in detailed treatment. Each vehicle is identified and its position, speed, acceleration, and other characteristics are stored. Microscopic models deal with two separate areas: streets and intersections. On streets, vehicle behavior is usually approximated by controlling one vehicle's speed by the preceding vehicle (car following). Car-following rules can be quite complex. Queueing behavior and the delay in accelerating away from the head of the queue are usually treated in detail; there are indications that this can be a critical factor in the accuracy of simulation, at least for urban networks. Lane changing may or may not be allowed, and other details of street traffic can be represented. Buses and trucks can also be handled separately.

Intersection behavior is considerably more complex and difficult to model. Problems of pedestrian interference, turning radii, and collision avoidance all must be faced. Usually these problems are treated superficially; for example, random delay is assigned for pedestrian interference. Because one primary function of network models is to evaluate signal control strategies, traffic signals are usually modeled in considerable detail.

The various simulation models (see Table 1) discussed here should be assumed, unless specifically indicated otherwise, to have the following characteristics. They are fully microscopic, meaning that the model stores a specific location, speed, and acceleration for each individual vehicle. Moreover, the vehicles engage in car following; that is, the speed of a vehicle is determined in some reasonable, predetermined manner by the speed of any vehicle directly in front of it. If there is no vehicle in front, it travels at its "desired" speed. Vehicles obey traffic signals and stop and yield signs.

All variables in the system (vehicle locations, speeds, accelerations, and signal indications) are updated once each time step, which is a user-specified constant, often 1 s. The computer program is written in FORTRAN. The simulation starts collecting data only after some initial period, which starts with the system empty and ends when some test for equilibrium is satisfied. Vehicle routes are determined from origin-destination tables (O-D) or the probability of turning left, through, or right at the end of each link (turning movements).

**OBSELETE MODELS**

The DISCRETE AND CONTINUOUS VARIABLE SIMULATION MODEL, developed by Mathewson, Trautman, and Gerlough, was perhaps the first traffic simulation model to actually be constructed (3, 4). It represented a signalized intersection network with one lane in each direction and pedestrians who delayed turning traffic. Validation was primarily by reasonableness. The model was based on an analog rather than a digital computer and therefore had to be rebuilt for each new intersection or network.

The STARK/NBS MODEL, produced by Stark of the National Bureau of Standards for the IBM 704 computer (5), achieved realism through the use of fairly detailed traffic behavior rules. The model included traffic signals, stop signs, one- and two-way streets, and oblique intersections within the program structure. Documentation for this model is hard to obtain and would not be sufficient to recreate it for use on another computer. The model provided complete statistics on vehicle behavior. The most interesting output was a movie of the simulation, which was the only verification of the realism of the model.

A simulation model was developed by Francis and Lott for the ROAD RESEARCH LABORATORY to examine proposed changes in traffic signal patterns in central London (6). This program had a moderate level of detail but did not distinguish between different types of
vehicles or follow them through the network. Each lane was divided into short sections that corresponded to a computer storage "bit." For each occupied section, a "one" would be represented in the storage location. Once moved from bit to bit in the computer memory to represent vehicle motion. This model simulated only signalized intersections of three or four legs. However, separate specification of right- and left-turning volumes and signal offsets and phasing was possible. Output included total time simulated, volumes and delays by link, average delays and average queue by link, and maximum queue by link. The maximum network size was 30 intersections, 80 links, and 20 peripheral arms. The program was executed at approximately 18/h times real time, where n was the number of intersections.

The **AUSTRALIAN OR PAK-POY MODEL** (7) was written to analyze traffic signal controllers and intersection capacity and was reasonably flexible. Validation was limited and consisted primarily of comparison of model results with before-and-after field studies of intersections. Documentation is unavailable. The primary outputs of the model were delay, queue length, and degree of saturation.

**VEHICLE TRAFFIC SIMULATOR (VTS)** was constructed by A. H. Blum (8, 9) and used general purpose simulation system (GPSS) language intersection modules assembled to form a network. The level of detail in VTS was good; the model simulated traffic signals and special sources and sinks of traffic. Fixed time signals were assumed to control all intersections, and the car-following logic was "change lane or assume speed of leading vehicle." Vehicles were not identified by type, but different types could be approximated by assigning vehicle characteristics. No validation was performed on this model and documentation is limited to one article (8).

**TRAFFIX** (simply a name) **MODEL** developed by Gerlough, Wagner, Rudder, and Katz, was the first network simulation model to enjoy wide usage and validation. At least four versions were produced from 1962 to 1968 (10, 11, 12, 13, 14). Although TRANS is now obsolete, it led to several other models (see the DYNEx, SIGNET, and UTC5-1 models below). The level of detail in the TRANS model was moderate. Automobiles were grouped
into short platoons. Vehicles could switch lanes in- 
stantaneously if they were queued behind left-turning 
vehicles. Vehicle behavior at signals was quite de- 
tailed. Different models of TRANS used different time 
steps; version 4 had a user-specified updated period 
that could not be less than 2 s. The TRANS model was 
the first to be widely validated, and validation runs 
were made in Washington, D.C., Los Angeles, and 
Detroit. The documentation explained fairly clearly 
how to prepare input data, but documentation of pro-
gram logic was scanty. Data preparation was tedious, 
because many parameters had to be supplied. Output 
gave the usual traffic parameters both link by link and 
networkwide.

HARTLEY, in England, developed a basic model utili-
ing a special purpose computer identified as the Atlas 
Digital Traffic Simulator. A diffuser was used to link 
intersections and replicate platoon behavior, while a 
random pulse generator created input traffic. Because 
of its fixed program, the model was relatively easy to 
runt but had very limited flexibility. Validation was by 
sensitivity analysis and reasonableness. Documentation 
was limited to two articles and a doctoral disserta-
tion (15, 16, 17).

The BIRMINGHAM MODEL, also developed in England, 
by Storey (18), handled four categories of vehicles. Car-
following logic was limited, and vehicles were assigned 
to their lanes based on their turning movement at the 
subsequent intersection. An unusual feature of this 
model was the assignment, to avoid path looping, of 
complete vehicle routes from entry through the network. 
The model could represent both fixed-time and vehicle-
actuated signals. Output was limited to printouts of 
vehicle location within the network and both journey 
times and delays for each route. Although the model 
was not validated, sensitivity testing indicated that it 
produced reasonable variations in output values based 
on input changes.

The DYNAMIC NETWORK ANALYSIS OF URBAN TRAFFIC FLOW 
(DYNET) SIMULATION MODEL was developed by Lieberman 
of General Applied Science Laboratories (GASL) in 1969 
(19). It was based on the TRANS model, used many of 
its features, and made numerous specific improvements. 
Particular effort was expended on making the model 
fully microscopic and simplifying the input requirements. 
Tracks and automobiles were represented separately 
and were generated according to a shifted exponential 
distribution. A vehicle was assigned a lane when it 
first entered a link according to the turn the vehicle 
expected to make when it exited from that link. A ve-

cle could change lanes when it entered a queue if the 
queue in the adjoining lane were shorter and the turning 
movement permissible from the new lane; the vehicle 
would change lanes only if an acceptable gap were avail-
able. Left-turning vehicles examined the oncoming 
traffic and remained in queue until a suitable gap ap-
ppeared. At stop signs and signals vehicles behaved 
very much as they did in the TRANS model. Its de-
tailed microscopic character and the shorter time step 
generally used (1 s) made the DYNET model actually 
run more slowly than the TRANS model. If the time 
steps had been the same in both models, speed of 
execution would have been comparable.

SAKAI AND NAGAO of Japan developed a simple platoon 
microscopic model (20). In this model, roadways were 
partitioned into 50-m segments. The computer kept 
track of the number of vehicles in these segments, and 
the average speed of these vehicles was a function of 
number of vehicles in the segment. Buses and trucks 
were replaced by their equivalent number of passenger 
automobiles.

CURRENT MODELS

VEHICLE TRAFFIC SIMULATION MODEL (VETRAS) (21) by IBM 
built on Blum's earlier VT model; it is somewhat 
more refined and detailed than models built from scratch. 
The logic includes provision for right turn on red, and 
car-following logic is "switch lanes or assume speed of 
leader car." The model allows easy modification of con-
trol logic and its modular structure and input format 
make it easy to set up a variety of networks. Network 
flexibility is limited by a restriction to four-legged in-
tersections. A variety of vehicle characteristics can 
be generated, but the addition of trucks and buses as 
distinct entities would require minor programming changes 
to produce separate statistics. Documentation for 
VETRAS is not extensive but is clear and does include 
flow charts. Validation was limited to sensitivity testing.

VEHICLE PERFORMANCE IN TRAFFIC (VIT) MODEL de-
veloped by the Aerospace Corporation is an excep-
tionally detailed, totally microscopic network model 
(22, 23, 24). Automobiles, trucks, and buses are gen-
erated according to a Poisson distribution. In addition 
to the individual vehicles, the characteristics of the 
drivers are generated stochastically and include desired 
speed, desired lane, gap acceptance characteristics, 
and a frustration factor that determines how long a 
driver will tolerate following a slower driver. These 
characteristics are correlated so that a reckless driver 
generally has the characteristics associated with that 
description. Vehicles follow each other according to a 
reasonable car-following law based on the apparent rate 
of change of the visual angle subtended by the leading 
vehicle. Lane changing is more complex in this model 
than in any other; a driver can change lanes simply because 
he is tired of following a slower driver. This is the 
only simulation program that includes "actual" acci-
dents. When two vehicles merge into the same spot, 
they are considered disabled and remain parked in that 
spot throughout the simulation. Flexibility of this model 
is somewhat restricted because surface streets are as-
sumed to intersect at right angles only. Also, there 
are some restrictions on traffic signal displays that 
make these representations less flexible than those found 
elsewhere. The model does not consider pedestrian 
interference. Validation of this model is poor; it has 
given reasonable results in one or two small tests. 
User documentation is not generally available. Input 
requirements are quite extensive because of the detailed 
microscopic nature of the model. The user may choose 
the desired output from a wide variety of traffic-related 
measures such as average speed, average delay, fuel 
consumption, and vehicle emissions.

URBAN TRAFFIC CONTROL SYSTEM (UTCS-1) AND SIMUL-
ATION OF CORRIDOR TRAFFIC (SCOT) are closely related 
models. UTCS-1 was developed by Peat, Marwick, 
Mitchell and Company and GASL, while SCOT was 
produced by GASL alone. UTCS-1 was designed to 
evaluate traffic signal systems (25, 26). It is based on 
the DYNET model and is fully microscopic: vehicle data 
are stored in packed words. The packing operations 
are done in the program where the Information is used 
or produced. Initialization continues until the number 
of vehicles in the system appears constant. The time 
step is fixed at 1 s. SCOT is identical to UTCS-1 except 
that the Dynamic Analysis of Freeway Traffic (DAFT) 
model has been included (27). On freeways, vehicles 
are grouped into platoons; individual statistics are not 
kept for each vehicle. The update period on the freeway 
is variable but is usually about 6 s. Level of detail in 
the UTCS-1 model is excellent. Pedestrian interference 
is represented, and the lane-changing rules are rea-
sonable. Oncoming traffic interferes with left turns as
does traffic that is backed up from the preceding intersection. The freeway portion of SCOT is less realistic. The speed rule as a function of density is somewhat arbitrary, and platoons have an inherent lack of realism. Validation of these models is moderately good. UTC-1 was extensively validated in Washington, D.C., and the freeway portion of SCOT was briefly evaluated in Dallas. UTC-1 has been widely used and is generally accepted as giving reasonable results. Documentation is hard to follow but is complete and readily available. The detailed character of the required data makes input preparation difficult and time consuming. The SCOT freeway data are partially awkward in that two parameters determined by a separate computer program are required. Output is full and complete both link by link and systemwide. A new version of UTC-1 gives emissions and fuel consumption.

The SIGNET (a signal network optimization system) model (28, 29) was developed by Davies as a master's thesis at Purdue University. Input to SGNIT is fairly complex. Output statistics include total vehicle miles, total delay, average delay, delay standard deviation, and average speed. The program will accommodate up to 85 links without modification of dimension statements. SGNIT borrows heavily from the TRANS model but is fully microscopic and utilizes computer words to describe vehicles in a manner similar to UTC-1. Stop and yield sign control is not represented. However, the model is fairly flexible in terms of intersection geometries and traffic controllers. Model validation was extremely limited and merely confirmed that the simulation outputs were reasonable and consistent with travel time studies. The documentation consists of a very thorough master's thesis project report that includes subroutine descriptions and flow charts.

MICRO-ASSIGNMENT (30, 31, 32), by Creighton, Hamburg, Inc., is a traffic assignment tool developed for transportation planning purposes; it was not designed as a traffic simulation program. However, microassignment considers origins and destinations on a block-by-block basis and therefore has many of the features normally associated with a simulation model. Basically, it was designed to give the expected traffic on any street for a given O-D pattern, but it also outputs the average speed and delay on each link. Microassignment is very macroscopic (for a simulation model) and ignores many of the fine details critical to traffic engineering. Its treatment of traffic signals includes only the coarsest representation of phasing and ignores the effects of signal offsets altogether. The key to the operation of microassignment is in the novel link-node structure. Nodes are placed in the center of each block so that a left-turning vehicle travels on a totally different link from one traveling straight through the intersection. In this way, each link can be assigned a fairly accurate value for the delay associated with the corresponding movement. These delays reflect the character of the control at the intersection (for example, a left-turn link has a lower delay if there is a protected left-turn phase) and, possibly, the traffic on opposing links. For any given O-D pattern, microassignment computes the minimum path when assigning all the traffic from a batch of origins to their destinations based on minimum time. A new minimum time is then computed, and another batch of trips is assigned. The batch size can be specified by the user; usually, to achieve maximum accuracy, only one path is loaded at a time. Because of its microscopic character, microassignment can rapidly handle very large networks of even 1000 city blocks. The assignments are essentially static, and a time compression ratio is not directly pertinent. A network of 1334 links took from 108 to 385 s (depending on the batch size) to do one complete assignment. If new assignments are required every 15 min, this corresponds to a time compression of 2.5:1 to 9:1.

CORQ for traffic assignment with queuing in corridors is a new traffic simulation model written by Yagar of the University of Waterloo (33). It has many features of microassignment. Given a set of O-D tables for, say, successive 15-min periods, it calculates the traffic assignments for one period. It uses a conventional link-node representation of the network and represents signals by state transitions. The unique feature of this model is that, when capacity cannot accommodate demand, the excess vehicles are stored on the link and added to the demand for the next period. One may visualize the flow as a fluid leaking through the network. User documentation is not yet available for CORQ.

SIGNAL OPTIMIZATION PROGRAMS

TRANSYT (simply a name) was developed by Robertson of the U.K. Transport and Road Research Laboratory as a signal optimization program (34, 35, 36, 37, 38) and, therefore, lies outside the primary emphasis of this paper. However, it contains, as an integral element, a simulation program that can be used without the optimization feature. The logic of the TRANSYT simulation program is deceptively simple. The program is totally macroscopic and completely deterministic; no random numbers are used at all. Uniform vehicular flow enters the upstream end of the most upstream link of the network. The flow arrives at the link's downstream end, where it accumulates during the red phase. When the signal turns green, vehicles discharge at the capacity rate of the signal until the queue is dissipated; thereafter the vehicles discharge at the rate at which they arrive. This emergent platoon of vehicles now has a specific shape and arrives at the next downstream stop line with a delay appropriate to the length of the link and to the speed of progression on the link. To enhance realism, the shape of the platoon is changed slightly to reflect dispersion. Again, the discharge from the signal is at intersection capacity until the queue is discharged. In this way, the shape of the platoons at any intersection reflects the effects of all the upstream intersections. Provisions are made for turning vehicles and for the arrival of vehicles at the stop line from the secondary flows that have turned onto the link. Thus TRANSYT is able to represent the performance of the network in a single pass without any initialization. Input preparation is comparatively simple and easy, but output, primarily delays and stops, is not as detailed as that of most models.

SIGNAL OPTIMIZATION (SIGOP I) (39, 40, 41), developed by Lieberman and Woo, is a descendant of TRANSYT and SIGOP I (SIGOP I is an optimization program now obsolete and contains nothing relevant to the subject of this paper) and, like them, is a signal optimization program. Like TRANSYT, it contains a macroscopic traffic flow model that can be used to evaluate the stops and delays of an existing signal system. The logic of SIGOP II utilizes dynamic programming techniques. SIGOP II orders the traffic flows into nine primary and secondary platoon combinations according to when the platoon departs from the first intersection and when it arrives at the second. Modification of platoon characteristics depends on the length of the link and the speed of progression on the link. Provision is made for turning movements and turn pockets. One evolutionary step in SIGOP II is its ability to model the effects of multiphase signals. Input preparation is fairly straightforward.
once the notation for describing multiphase signals has been mastered. Special coding sheets and a fully documented case study are included with the documentation. Output includes time-space plots of signal control along specified arterials as well as link-by-link statistics. SIGOP II is still being field tested by the Federal Highway Administration and, therefore, is currently available only on an experimental basis.

CONOID is a corridor optimization program for queuing (42, 43). As such, the range of networks that can be analyzed is restricted; the program explicitly assumes a one-directional freeway and a parallel arterial with two-way connecting streets. The freeway part of the program is FREQSC (44), which was developed by May and his coworkers at the University of California. The surface street portion is the TRANSYT program discussed above. Both optimization programs contain simulation programs as subsets. If simple simulation is desired, the network geometry can be considerably more general than that assumed for optimization. Both models have been well validated.

REFERENCES

Simulation of Bus Lane Operations in Downtown Areas

R. W. Bowes and J. van der Mark, De Leuw Cather, Canada, Ltd.

An investigation of potential operational capacities of downtown bus lanes was carried out in Ottawa, Canada. The results could improve coordination between bus priority treatments within downtown areas and those on the major transit corridors feeding the downtown areas. This paper describes the simulation model that was developed during the course of the study to simulate bus behavior in a downtown bus lane. Several operating strategies were tested and evaluated. Depending on the strategy selected, a bus flow rate of 150 to 170 buses/h can be achieved. This flow rate can accommodate 8000 to 9000 passengers/h with acceptable loading standards, operating speeds, and existing standard equipment. Further tests and research are suggested.

Bus priority systems, introduced in many cities, may utilize different measures from the groups listed below:

1. Freeway related measures: reserved lanes, separate roadways, exclusive ramps, queue jumping, metered collection, and bypass of toll barriers;
2. Arterial related measures: reserved lanes with flow or contra-flow, separate roadways (bus-only streets), special signalization, special turn permission, and pulling away priority at bus stops;
3. Busway measure: exclusive bus-only roadway either grade separated or with at-grade intersections; and
4. Terminal measure: off-street loading, unloading, and short-term storage of buses, usually connected to freeways or arterials with bus priority measures.

Coordination of the operational capabilities of these various measures could lead to an improved level of service and more efficient use of buses during peak periods. For example, if downtown bus lanes can accommodate the large number of buses arriving from reserved freeway lanes from suburban areas, overall travel times can be reduced. Generally accepted minimum installation criteria for bus lanes are 30 to 90 buses/h (4); maximum operating capacities of downtown bus lanes are currently 90 to 120 buses/h. These are sometimes exceeded, as in Ottawa, where flows in excess of 140 buses/h occur in some bus lanes.

The success of bus transit in this context relies on how far into the future a municipality’s improved bus-based system can meet peak-hour transit needs in major demand areas. The capacity limits of a bus-based system will probably be experienced in the downtown collection and distribution system rather than on reserved freeway lanes and busways between downtown and suburban areas. Because bus lane volumes in Ottawa are already as high as any that have ever occurred in the transit industry’s experience, a study was commissioned to investigate the operating capacities of bus lanes in a downtown environment.

After briefly reviewing the literature and summarizing field experiments, this paper will describe the simulation model we developed during the course of the project. A number of bus operating strategies were tested and...