



**Transportation Research Forum**

---

A Review of Dynamic Traffic Assignment Computer Packages

Author(s): Mansoureh Jeihani

Source: *Journal of the Transportation Research Forum*, Vol. 46, No. 2 (Summer 2007), pp. 35-46

Published by: Transportation Research Forum

Stable URL: <http://www.trforum.org/journal>

---

The Transportation Research Forum, founded in 1958, is an independent, nonprofit organization of transportation professionals who conduct, use, and benefit from research. Its purpose is to provide an impartial meeting ground for carriers, shippers, government officials, consultants, university researchers, suppliers, and others seeking exchange of information and ideas related to both passenger and freight transportation. More information on the Transportation Research Forum can be found on the Web at [www.trforum.org](http://www.trforum.org).

# A Review of Dynamic Traffic Assignment Computer Packages

by Mansoureh Jeihani

*This paper reviews the dynamic traffic assignment models in some well-known computer packages. It describes demand estimation, supply presentation, methods for computing dynamic user equilibria, and convergence among these packages with a concentration on TRANSIMS. TRANSIMS tries to take advantage of both precision and computational speed by estimating second-by-second movements of individual travelers and applying parallel processing and cellular automata methods. Although TRANSIMS addresses some of the existing problems in dynamic traffic assignment models, more studies on it and improvements in it are still needed.*

## INTRODUCTION

Traffic assignment is the allocation of origin-destination (O-D) flows to transportation routes, based on factors that affect route choice. There are two different approaches for assigning traffic: the system optimal (SO) and the user equilibrium (UE) approaches. The SO approach minimizes the total system travel cost (i.e. travel time), and the UE approach seeks user path assignments in which no single user can improve his/her experienced travel time by unilaterally switching routes. Traffic assignment could also be classified into two major categories: static and dynamic. Static traffic assignment models assume that link flows and link travel times remain constant over the planning horizon, while in dynamic traffic assignment (DTA) models, the link flows and link travel times are allowed to be time dependent. DTA models find time-dependent link flows in a transportation network.

Although static equilibrium models are adequate for long-term planning analyses, they fail to capture the essential features of traffic congestion. Many researchers have attempted to develop DTA models for more than three decades. Nevertheless, these models, which are an extension of the static traffic assignment models, are still relatively unexplored and lack a proper formulation due to their complexity as compared with the static models. Therefore, developing DTA models that are applicable to large-scale transportation networks is still of interest to researchers. A review of the literature in this area is outside the scope of this paper, but is presented in Peeta and Ziliaskopoulos (2001).

While the static assignment method still dominates most transportation planning exercises, it has been observed that the DTA will replace the static assignment soon. Such a trend can be verified by the fact that most transportation planning software vendors have started developing DTA-enabled planning tools.

This paper reviews DTA models in some of the existing computer packages. Although the focus is on planning, some packages that address DTA at the operational level are also briefly reviewed. We explain Transportation Analysis and Simulation Systems (TRANSIMS) more extensively for two reasons. First, in contrast to the other packages, TRANSIMS has not received good exposure and its capabilities are unknown to many researchers in the transportation field. Second, the author has extensive experience with TRANSIMS (Jeihani et al. 2006a; Jeihani et al. 2006b; Hobeika et al. 2002).

DTA packages typically use a traffic simulator to capture the complex dynamic interactions among vehicles on a network. They move vehicles on the network based on different formulas and defined relationships between vehicle speeds, road capacity, and flows. These simulation-based

models are categorized into three groups: macroscopic, mesoscopic, and microscopic models. Macroscopic models use aggregate equations and therefore provide only gross approximations. Although they partially address the traffic modeling needs of freeways, a greater level of detail may be required for surface street networks, especially at intersections. Mesoscopic models use macroscopic equations, but move vehicles in small-size packets. On the other hand, microscopic models keep track of vehicles second by second and track details of individual drivers and vehicles, and the interactions between them. They are more accurate than the mesoscopic models, but entail large computational requirements. There is a tradeoff between precision and computational speed in utilizing mesoscopic or microscopic models.

The packages reviewed in this study are TRANSIMS, PARAMICS, and VISSIM, which are microscopic; and DYNASMART, DynaMIT, and CONTRAM, which are mesoscopic. Different aspects of the DTA model within the planning context, such as demand estimation and prediction, transportation supply, the methodology to find DTA, equilibrium, and convergence, are reviewed in these packages.

In the microscopic packages, TRANSIMS is developed completely at the planning level, while PARAMICS and VISSIM are developed fully at the operational level. However, VISSIM has the accompanying package VISUM, which provides planning-level tools. Since VISUM provides static dynamic assignment, it is not explained in this study. In the mesoscopic packages CONTRAM is developed at the planning level but can be also used at the operational level. DYNASMART and DynaMIT were developed for operational purposes and have a planning version as well. We focus on reviewing the planning versions of these packages, DYNASMART-P and DynaMIT-P.

## DEMAND ESTIMATION AND PREDICTION

In planning applications, the most challenging work is to estimate and predict the time-dependent origin-destination (O-D) demand matrices. These matrices, which are used as an input in traffic assignment models, are computed within a demand estimation modeling framework. Existing demand models do not provide satisfactory methods for estimating time-dependent O-D demands. While there are a few studies related to time-dependent demand estimations as stated in Peeta and Ziliaskopoulos (2001), there is no framework in the literature to produce time-dependent demand data. The existing models usually extend the static demand estimation models. The most commonly used static demand estimation models are trip generation and trip distribution in the four-step models. The four-step models determine the 24-hour O-D matrix and then separate it by the time of day into peak and off-peak periods. The approach adopted by TRANSIMS for demand estimation and prediction is different from that of other models. It introduces a framework to produce time-dependent demand data for individual travelers for a 24-hour period using census data, land-use data, and an activity survey of travelers.

All packages but TRANSIMS rely on external demand estimation tools. Since there is no realistic time-dependent O-D demand (just a simple extension of the static models, as explained before) to be fed into these packages, the reliability of their results is questionable. However, in some of the packages, it has been attempted to increase the reliability of the results by improving the O-D demand input. DynaMIT-P modifies the time-sliced O-D demand to reduce the effect of the unrealistic O-D demand input on the traffic assignment results using a demand simulator. The demand simulator is a microscopic simulator of traveler behavior in terms of route choice, departure time choice, and response to information. It estimates and predicts O-D demand based on historical information and the drivers' response to information. Since the demand simulator is microscopic, the historical O-D matrices are disaggregated into a historical population of travelers. A vector of socioeconomic characteristics is generated by Monte-Carlo simulation based on their distribution within the actual population and is assigned to each traveler. In contrast to other packages where demand is a time-sliced matrix of O-D flows, demand in TRANSIMS is activity-based as explained below.

TRANSIMS proposes a time-dependent activity-based demand modeling. Since TRANSIMS is based on the movement of individual travelers between activities at different locations, it creates a synthetic population that represents every individual on the network. TRANSIMS uses an algorithm developed by Beckman et al. (1996) to generate synthetic populations from available census data for a given planning year, using the Population Synthesizer module. Land-use data are then used to place individual households at activity locations along the transportation network, where an activity location represents a place that a household member would travel to and from. TRANSIMS uses activity location rather than zones to represent travel. One zone could contain several different activity locations. These activity locations could be residential areas, business places, or schools. Non-home activity locations reflect land-use and employment data obtained from the metropolitan area.

The Activity Generator module generates a list of activities for each member of a synthetic population. To create the list of activities, the Activity Generator performs the following: (a) creates skeletal activity patterns from the survey of households; (b) uses a Classification and Regression Tree (CART) algorithm to build a classification tree based on household demographic data; (c) matches each synthetic household with a survey household, (d) generates activity times and durations, and (e) generates non-home activity locations.

TRANSIMS uses the CART algorithm to produce a classification of household demographic characteristics, which are designated as independent variables, based on household travel behavior indices serving as dependent variables. Each survey household is effectively classified by the CART algorithm into one of the tree's terminal nodes representing the household demographic characteristic. This tree is sensitive to household travel behavior characteristics but is parsimonious with respect to household characteristics that do not affect travel behavior.

Then, a discrete choice model is used to select an appropriate place for each activity. Using a variety of activities such as home, work, school, college, shop, visit, serving passengers, and other trip chains between activities is suggested in TRANSIMS. Each individual in the household has a chain of activities; and each activity consists of an activity type and its priority, a starting and ending time preference, a preferred mode of transportation, a vehicle preference (if appropriate), a list of possible locations for an activity, and a list of other participants (if the activity is "shared-ride"). The set of activities for each household is based on the household's demographics. Consequently, TRANSIMS develops a time-dependent activity-based demand estimation and forecast. This procedure is space intensive but not very computational intensive.

## **TRANSPORTATION SUPPLY AND OUTPUT**

Transportation supply refers to the transportation network and its associated traffic control systems. Since most existing models and packages provide aggregate O-D demand information based on interchanges between different zonal centroids, their network structure utilizes zonal centroids along with virtual links that connect these centroids. These virtual links are usually called centroid connectors. The definition and size of traffic zones is critical in determining the realism and accuracy of the traffic model. Since TRANSIMS uses activity locations within the street network instead of zones, all roads in the network from small lanes to freeways are represented, and no centroids or centroid connectors are introduced.

In macroscopic and mesoscopic models, transportation networks are presented as nodes and links representing intersections and road segments. VISSIM actually uses a link/connector network representation rather than a link/node. The link characteristics that are presented by all packages include link length, capacity, speed, number of lanes, and link type. The microscopic models provide detailed information about streets, intersections, signals, and transit modes within the road network. Furthermore, detailed network characteristics that exist in the real world such as pocket lanes, merge lanes, parking spots, bus stops, link grades, and lane use restrictions are all incorporated in these packages.

Signal cycles and random variations in traffic cannot be modeled explicitly in CONTRAM. Although DynaMIT-P and DYNASMART-P account for signal control, intersection movement delay, and ramp metering, they might not be very precise due to the mesoscopic modeling paradigm. This capability is accommodated more accurately in microscopic models.

Microscopic models provide a complete picture of traffic in a fully described network in which all results are derived from the microscopic model. PARAMICS allows the user to build the network on top of a road geometry file. VISSIM allows for freehand drafting of the network as well as using air photographs. TRANSIMS produces traveler events, spatial and temporal summary data, and snapshot data. Analysts can specify the simulation period, the portion of the network to be simulated, the sampling rate, and the reporting frequency.

## **TRAFFIC ASSIGNMENT AND EQUILIBRIUM**

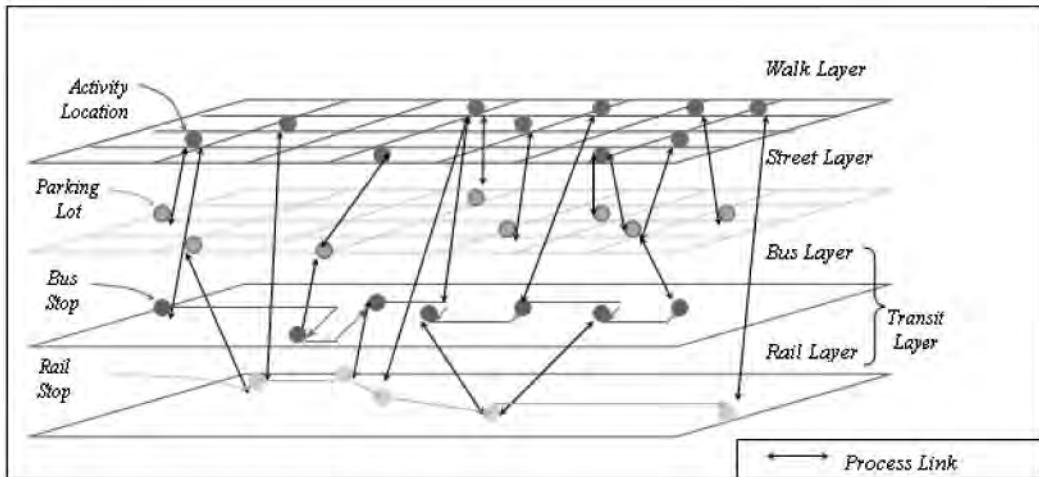
### **TRANSIMS' Traffic Assignment Model**

TRANSIMS performs the dynamic traffic assignment using the Route Planner, the Microsimulator, and the Feedback modules. To assign demand on the network, shortest-path algorithms are used. The shortest path could be the shortest time path or the shortest cost path based on any suitably defined cost function. The shortest-path algorithm utilized in the Route Planner module of TRANSIMS solves a time-dependent label-constrained shortest-path problem. Here the labels refer to travel modes such as car, walk, and rail transit, and a string of admissible labels is also specified as a constraint when computing the time-dependent shortest paths. The solution method used is an extension of the classical Dijkstra's algorithm developed by Barrett et al. (2001). This uses non-deterministic finite automata (NFA) to create a composite graph consisting of all possible combinations of paths that satisfy the specified label constraints for the trip from an origin node to a destination node.<sup>1</sup>

Each link has a travel time associated with it, which can be constant as for example for walking, or might be dependent on the time of day, as with street links, and railway links. The default travel time for a link is the free-flow travel time. More realistic travel times are calculated by the Microsimulator which represents the average delay experienced by vehicles traversing a link over 15-minute intervals. Therefore, in the traffic assignment process, TRANSIMS divides the time horizon into 15-minute time-slots. The label-constrained time-dependent shortest-path algorithm finds the shortest path for each traveler based on the arrival time at each node and the link travel time values for the relevant time slot.

Besides time dependency the problem is multi-modal and requires the determination of a shortest path based on permissible strings of travel modes as specified by the label constraint. Computing shortest paths on time-dependent multi-modal networks is challenging and very little effort has been devoted to this problem in the literature. Most existing models assign paths to vehicles rather than to travelers, and so mode-split is typically used. Some researchers such as Marcotte and Wynter (2004), however, attempt to solve time-dependent multi-modal shortest path problems.

TRANSIMS accommodates the modeling of 12 modes: Walk, Bike, Car, Bus, Light Rail, Regional Rail, Rapid Rail, Trolley, Street Car, Transit, Magic Move (School Bus), and Magic Move (Other). The Magic Move mode is an unrouted mode that is used to enable the use of travel modes that are not explicitly supported by the Route Planner or the Microsimulator such as school-bus mode. To support finding multimodal shortest paths in networks, the Route Planner views the network as a set of interconnected, single mode layers. Special links, called process links, are used to connect one single mode layer to another. Hence, intermodal transitions are allowed using process links. Figure 1 depicts the concept of the various layers and the associated process links.

**Figure 1: A High-Level Depiction of the Various Layers used by TRANSIMS**

Source: Hobeika et al. (2002), Chapter 1, Page 27

To maintain computational efficiency, a heap implementation is used by the algorithm.<sup>2</sup> Also, the network is converted to an internal route network that represents a weighted directed graph. To enhance performance, the Route Planner examines distance-based delays to focus the determination of paths that more pointedly lead toward the destination.

Some modeling as well as algorithmic enhancements of the above shortest path algorithms have been done by researchers at Virginia Tech. Sherali et al. (2003) developed an effective method for implicitly working with the composite graphs rather than constructing the full composite graphs a priori. This model is based on the partitioned shortest-path algorithmic developed by Glover et al. (1985) and its dynamic programming (DP) interpretation developed by Sherali (1991). Furthermore, to reduce the search effort, Sherali et al. (2003) proposed several heuristic curtailing schemes by focusing the search to systematically proceed toward the destination from the origin while avoiding the searching of paths that are beyond a defined boundary. They developed an *Ellipsoidal Region Technique* (ERT) that examines only those paths within the union of some three defined ellipsoidal regions that envelope the origin and destination, including any freeway sections and related entrance and exit ramps. This ERT technique effectively curtailed the search and was demonstrated to find solutions within 7% of optimality while saving 33.57% CPU time as compared with the exact method on some test cases related to the Portland, Oregon, test network.

Sherali et al. (2006) also developed an approach-dependent, time-dependent, label-constraint shortest-path model to accommodate turn-penalties in transportation networks, where the time spent at an intersection before entering the next link depends on whether the driver travels straight through the intersection, makes a right turn, or makes a left turn. Further, heap-based implementations and refined ERT-based heuristics were shown to determine solutions within 0.78% of optimality while curtailing effort required by the exact method by 56.77%.

The traffic assignment problem in TRANSIMS is solved by interfacing the Activity Generator, the Route Planner, and the Microsimulator modules using the Feedback module, which can be designed by the analyst by writing some scripts in Perl programming language.<sup>3</sup> However, the coding is simply to determine which modules should be used, which order they should be run, and to specify the convergence criteria specifics. Feedback loops can be performed on any two modules to calibrate the models and assign and stabilize the traffic. The traffic assignment in TRANSIMS can be executed using any combination of departure time choice, mode choice, and route choice models. This ability is unique, to the best of the author's knowledge. The simplest version of the traffic assignment process is only a route choice model and it is applied to the Portland study (2001);



it is a loop between the Route Planner and the Microsimulator modules. In the first run, the Route Planner uses free-flow speeds on each link to estimate travel times and to find shortest paths. This is not realistic, because there are other vehicles on the link and speed is not equal to free-flow speed due to congestion. The Microsimulator produces the new travel times based on accurate vehicle speeds. The new travel times computed by the Microsimulator are fed back to the Route Planner, and the new routes are determined as the shortest paths for selected travelers. The Route Planner finds a time-dependent shortest path between different activities for each individual.

This iteration between the Route Planner and the Microsimulator is repeated until a stopping criterion is met. In the Portland study (2001), the selection of people to be re-routed was made by a uniform random sampling among all households. It is stated in this report that a targeted selection of travelers yielded results that were insignificantly better and in some cases worse than using a uniform random selection.

Nagel et al. (1998) presented results of simulation studies from the Dallas-Fort Worth area on a relatively large network with 6,124 links. They performed a feedback between the Route Planner and the Microsimulator to find a reasonable traffic assignment. They found that the best performing method was similar to the method of successive averaging (MSA), (Sheffi 1985). They started the rerouting fraction at 30%, and slowly decreased this to 5% by the 20<sup>th</sup> iteration. The stopping criterion used was the vanishing of gridlocks in the Microsimulation, which was realized by visualization.

Using free-flow speeds at the first iteration between the Route Planner and the Microsimulator causes heavy congestion on some links and it needs several subsequent iterations to clear this congestion. For the Portland study (2001), this congestion turned out to be too excessive to simulate. Hence, travelers were loaded incrementally onto the network. The iteration between the Microsimulator and the Route Planner progresses until a stopping criterion is met. In deterministic steady-state assignment, the process is terminated when changes are smaller than some pre-defined values and the method tends to converge to a near-equilibrium solution. Since micro simulations are neither deterministic nor steady-state, it is unclear if any described property holds true for the resulting solution. Another difficulty with the current method is that travel time is minimized with respect to the previous iteration's solution, which could induce excessive oscillations. Therefore, the current method needs to be improved.

To address some of the above problems, Raney and Nagel (2002) implemented a modified version of TRANSIMS for all of Switzerland. They found the shortest path for each traveler and incorporated a memory of past plans. Travelers selected their new plans based on the performance of the routes in their memories. Thus, a fraction of travelers were chosen to be re-planned based on informed decisions rather than by random selection. The selection methodology adopted is similar to that of logit models used in discrete choice analysis. Their test network comprised 10,572 nodes and 28,622 links, and 7.5 million travelers and 20 million trips were studied over a 24-hour period. The method terminated after 49 iterations, which was time consuming since the Microsimulator and the Route Planner are both quite computationally intensive.

In all of the above implementations, there are usually a large number of iterations performed between the Route Planner and the Microsimulator, which is time consuming. These excessive iterations are necessary to clear heavy congestion on some links due to the use of the free-flow speed to route travelers onto the network in the first iteration. TRANSIMS routes one traveler at a time but does not update link volumes and link travel times after each routing. Therefore, each person is routed regardless of the existence of other travelers on the network. After routing all the travelers via the Route Planner module, the Microsimulator simulates these travelers and their interactions on the network, and finds link volumes and link travel times. Then the Route Planner reroutes a proportion of travelers using the updated link travel times. This sequence of iterations between the Route Planner and the Microsimulator continues until a stopping criterion is satisfied. Another problem is that this stopping itself is based simply on visualization, which is imprecise.

Jeihani et al. (2006a) proposed an approach for determining dynamic user equilibria to improve the current DTA model in TRANSIMS. The method is a two-stage process that employs a combined

use of link performance functions and a micro simulator in order to design a framework suitable for application to real transportation systems. In the first stage, they used a new version of the Route Planner in which they route one traveler at a time and update the link travel times after each routing, using a link performance function. Therefore, the next traveler is routed based on the most updated travel time, thereby precluding heavy congestion on key links. This stage is executed only once to distribute travelers well on the network from the beginning. This obviates an excessive number of iterations between the Route Planner and the Microsimulator in order to re-distribute them. This also mitigates the oscillations that frequently occur in the present version of TRANSIMS. In the second stage, the modified process to find a near-equilibrium by alternating between the current Route Planner and the Microsimulator by iteratively redistributing a subset of travelers that have the highest ratio of the actual travel time (experienced in the Microsimulator) to the travel time that was calculated in the Route Planner. This method was applied to a large-scale network, Bignet, which is part of the transportation network of Portland, Oregon. The results exhibited an improved distribution of travelers while taking 17% to 33% of the effort required by the current version of TRANSIMS.

### **CONTRAM's Traffic Assignment Model**

The dynamic user equilibrium procedure adopted in CONTRAM divides the given time-sliced demand into small packets, routes them sequentially along minimum cost routes in the order of the trip start times, uses the time-dependent link-based all-or-nothing tree-building method, and then updates the link travel times using the Cost Benefit Analysis (COBA) speed/flow relationship (Taylor 2003). COBA is a set of continuous piecewise linear functions. These are static functions that modify link speeds in each time slice based on average link flow on that time slice. Time-dependent queuing theory is at the heart of CONTRAM. Queuing is modeled on the motorways using an explicit bottleneck for congested networks.

After routing all the packets, CONTRAM re-assigns them over several subsequent iterations, because each packet can influence the travel times of other packets. It re-assigns each packet by deducing it from the network, updating network states, and finding the shortest path for the packet. To calculate the shortest path, the packet is added temporarily to all links of each possible path between the origin and the destination. This means that a weaker definition of user equilibrium is used: switching a path is allowed only if this switch decreases the actual resulting travel time. The procedure is terminated after a user-specified maximum number of iterations, until the algorithm converges to an equilibrium solution, or until certain stability criteria are satisfied. Leonard et al. (1978) state that sensitivity tests revealed poor convergence properties for this process. To remedy this, they proposed a 'fixed route' option, in which particular traffic demands are forced to choose certain specific routes, however unattractive they might be.

The CONTRAM model is dynamic and time-sensitive to each routing. After re-routing each packet, network states such as link travel times are updated. Therefore, packets are adequately distributed onto the network using the most updated network states. Nevertheless, the method does not converge and the method to make it converge is unrealistic and deviates from user equilibrium principles. Most operations in CONTRAM adopt macroscopic traffic models which are imprecise; however the model provides a fairly efficient and flexible way of representing the real world. CONTRAM supports only the car mode in a network. Of the three models, route choice, mode choice, and departure time choice, only route choice is used in CONTRAM.

### **DynaMIT-P's Traffic Assignment Model**

DynaMIT-P is an offline version of the real-time traffic estimation and prediction that has been developed for planning purposes (Ben-Akiva 2001, DynaMIT 2005). It consists of a supply simulator, a demand simulator, and an algorithm to conduct the interaction between demand and



supply. The demand simulator is a microscopic simulator which was explained briefly above. The supply simulator is a mesoscopic simulator that captures traffic dynamics such as queues, spillbacks, and congestion. The choice of the level of aggregation of vehicles into homogeneous packets, and the choice of time steps determine the level of detail in the supply simulation. It is assumed the only mode is a car and the departure-time interval is the interval corresponding to each O-D matrix. A travel choice model is used to route travelers using socioeconomic characteristics such as the value of time, information source, and trip characteristics such as trip purpose. Demand estimation is a fixed-point problem whose solution is reached through an iterative process with a new assignment matrix computed at each step.

In DynaMIT-P, a list of travelers and their paths is generated based on their current travel times using a demand simulator that implements demand disaggregation. Then the aggregated supply simulator is run for the entire planning horizon to provide aggregate link travel times that are experienced by the travelers. The expected travel times (toward computing equilibrium solutions) are updated by a convex combination of the travelers' expected travel times and their recent experienced travel times. The above procedure is repeated until a convergence criterion based on the expected travel times of travelers matching their experienced travel times is satisfied. This is checked by comparing the equilibrium travel times in the current and the previous iterations using a weighted norm function.

The disaggregation procedure in the demand simulation is computationally intensive because it needs to assign characteristics and initial routes to all the drivers. The drivers need to be sorted based on departure time intervals. DynaMIT-P is a tool that can be used for the evaluation of Intelligent Transportation Systems (ITS) at the planning level and for the evaluation of short-term planning projects, rather than for long-term planning purposes. It models the day-to-day evolution of traffic, traveler behavior, and network performance for special events and situations such as incidents, weather emergencies, and sporting events.

### **DYNASMART-P's Traffic Assignment Model**

DYNASMART-P (Peeta and Mahmassani 1995, Mahmassani 2001) solves the dynamic user equilibrium problem and the dynamic system optimal problem for O-D demands having fixed departure times, using heuristic simulation-based iterative procedure. For the user equilibrium problem, DYNASMART-P initially finds the shortest path for the given O-D matrices using free-flow travel times to assign them onto the network. Then, it iteratively simulates vehicles and finds the new travel times and assigns travelers based on the new travel times using MSA. The shortest path algorithm applied in DYNASMART is a multiple user class K-shortest-path algorithm with movement penalties. Calculating K different paths for every O-D pair is computationally intensive, especially when the time-slots are small, whereby the number of O-D matrices (one for each time-slot) is considerably high. To enhance the model's performance, the K paths are not recalculated at every simulation time-step, but are recomputed only at pre-specified intervals. In the Knoxville, Tennessee, test network with 1,870 nodes and 3,304 links the computational time using a Pentium-4 1.4-GHz processor with 2.0 GB memory is reported to be 3 hours and 15 minutes when computing the two shortest paths (Peeta and Mahmassani 1995).

DYNASMART-P recognizes four vehicle types, namely, passenger cars (PC), trucks, high occupancy vehicles (HOV), and buses for their effect on traffic conditions (such as link capacity, speed, density, and volume) and consequently, on path assignments. Passenger cars, trucks, and HOVs are specified as fractions of the overall vehicle fleet. In this case, the specified O-D demand matrix reflects vehicular trips. Alternatively, the user may specify a separate O-D demand matrix to account for trucks in the network. If this is done, there is no need to specify the fraction of trucks in the overall vehicle fleet.

Although both DYNASMART-P and DynaMIT-P use heuristic iterative processes to re-assign travelers based on the updated link travel times in iterations there are some differences between

them. In DYNASMART-P the algorithm stops if the link volumes do not change in two consecutive iterations. Comparatively, DynaMIT stops when travelers experience the same expected travel time.

### **PARAMICS's Traffic Assignment Model**

PARAMICS (Quadstone 1999) models decisions and perceptions of drivers as well as physical characteristics of vehicles. It uses Driver-Vehicle Unit (DVU) as packets of travelers. The DVU can have information about the car such as type (length, height, and width), age, maximum speed, and information about the drivers such as aggressiveness, awareness, and O-D. It moves vehicles on a network through car-following and lane-change models.

PARAMICS supports three assignment methods: all-or-nothing, stochastic, and dynamic feedback. All-or-nothing assignment method assumes that drivers have the same information to choose their routes and congestion does not affect their decision. In stochastic assignment method, drivers are assumed to have different perceptions of travel costs. Dynamic feedback method assumes that drivers familiar with alternative routes will reroute if traffic conditions are provided to them.

PARAMICS uses an O-D matrix combined with a time-varying profile. This means that the demand on the network between each O-D pair can vary by time and can also vary relative to other O-D pairs. In the initialization process, a base routing tree is constructed. Routing trees are stored as tables of costs to the destination. At least two routing trees are constructed. The route choice decisions are dynamic, in that at each time a DVU knows the final destination and only the next N route choices, where N is a small number. Therefore, the procedure is fast and simple. Also, a table of information about the cost of each link is stored at each node for a given set of vehicle types, drivers (familiar or unfamiliar), and destination. These data are updated periodically if the feedback is switched on. Each vehicle traverses the network via the cheapest path using the tables at each node and selecting the least cost link at each node.

To model the dynamic traffic assignment, the user enables cost recalculation on a regular basis such as every five minutes of simulation time. This enables the mean simulated travel times for links to be used rather than free-flow travel times. The updated cost is used to create a new routing tree. Drivers are classified into familiar and unfamiliar drivers. Since unfamiliar drivers are not familiar with the area and don't know the routes, they use the signposted routes. Familiar drivers are drivers who know the area very well and use the routes regularly. Therefore, they have an estimation of travel times on each route and choose their routes based on this estimation. The route tree is recalculated only for familiar drivers.

### **VISSIM's Traffic Assignment Model**

VISSIM allows the user to have a variety of vehicles including cars, trucks, vehicles equipped with route guidance systems, buses, heavy rail and light rail, bicyclists, wheelchairs, pedestrians and even aircraft. VISSIM supports three assignment methods: fixed routes, dynamic routes, and dynamic traffic assignment. The basic method assumes that traffic is stochastically distributed over fixed routes from origins to destinations. Dynamic routes allow traffic to be assigned to user-specified paths when specific events occur. Dynamic traffic assignment in VISSIM assigns traffic to the network using O-D matrices (time and vehicle class-dependent) and travel costs. It uses the psycho-physical car-following model developed by Wiedemann (1974). Based on this model, a faster moving vehicle decelerates while approaching a slower vehicle and follows the slower vehicle with a safe headway. The user can set some of the car following model parameters to reflect a customized driver behavior model. A rule-based model is used for lane change to occur only if it is safe to do so. Also, a vehicle will change lanes if it is approaching a required turn or if the lead vehicle is traveling at a slower speed than the vehicle following is traveling.

VISSIM assumes that some travelers do not choose shortest paths. Therefore, it presents a set of  $k$  best routes but there are no efficient methods to compute this set. It computes the best path in each step and then archives it as  $k$  best paths. This archive grows during the iterative process. The choice of a route is determined using discrete choice modeling. Given a set of routes and their generalized cost, which is a combination of distance, travel time, and other costs such as tolls, the percentage of drivers that choose each route is calculated (PTV AG 2000).

In VISSIM, an abstract network representation is built and used for dynamic traffic assignment rather than using the detailed network. The iteration of the simulation runs continues until volumes and travel times on the links of the network do not change significantly compared to the previous iteration.

**Table 1: Dynamic Traffic Assignment Characteristics in Different Transportation Packages**

Package	Model	Convergence Criteria	R/M/D <sup>I</sup>	TB/ AB <sup>II</sup>	L/P <sup>III</sup>	S/M <sup>IV</sup>
CONTRAM	mesoscopic	User defined	R	TB	L	S
DynaMIT-P	mesoscopic	Expected travel time equal to experienced travel time	R/D	TB	P	S
DYNASMART-P	mesoscopic	Link volumes in the current iteration must be the same as in the previous iteration	R/M	TB	P	M
PARAMICS	microscopic	No new routing tree is made	R	TB	L	S
TRANSIMS	microscopic	User defined	R/M/D	AB	L	M
VISSIM	microscopic	Link volumes in the current iteration must be the same as in the previous iteration	R	TB	P	M

I: Route Choice/Mode Choice/Destination Choice used in dynamic traffic assignment

II: Trip Based or Activity Based

III: Link-based or Path-based routing method

IV: Single mode (car) or Multi-modal support

## SUMMARY AND CONCLUSION

Dynamic traffic assignment has been of interest to many researchers in the past three decades in both real-time traffic operations and in long-term planning. To capture the complex traffic flow dynamics, simulation-based models are extensively used. Most simulation packages have been developed for real-time purposes. Some existing packages and methodologies used to compute user equilibria are reviewed in this paper, and are contrasted with TRANSIMS.

TRANSIMS is a microscopic large-scale transportation system simulation framework that has been designed for planning contexts. It tracks individual travelers and estimates second-by-second

movements during a 24-hour period. It might be argued that modeling the real-world of traffic in detail would be computationally intensive. However, TRANSIMS tackles this problem by applying parallel processing and the high-speed cellular automata approach in micro simulation. Therefore, it simulates every event in a reasonable amount of time. Also, increased computational power of computers makes using such a microscopic model possible.

Unlike other packages that require the O-D demand as input, TRANSIMS estimates and predicts time-dependent demand. It treats individuals in a disaggregated fashion and it is multi-modal. Its traffic assignment supports route choice, mode choice, and departure time choice. It also does not need any user-class definition since it handles each person individually.

Some of the existing problems in using dynamic traffic assignment in planning contexts are addressed by TRANSIMS. However, the current dynamic traffic assignment routines it uses have stability and computational problems some of which have been addressed by researchers at Virginia Tech as discussed in this paper. Several improvements are still needed and it is hoped that this paper will serve to stimulate researchers to examine TRANSIMS more closely than has been forthcoming thus far.

## Endnotes

1. NFA consists of a set of possible states, a state transition function, and a set of accepting states. A composite graph is provided using NFA, and then Dijkstra's shortest path algorithm is applied to this composite graph.
2. Heap is a specialized tree-based data structure.
3. Perl is a dynamic programming language, which borrows features from various languages such as C. It is practical, easy to use, and efficient. Perl was originally developed for text manipulation but it is used in variety of tasks now.

## References

- Barrett, C., R. Jacob, and M. Marathe. "Formal-Language-Constrained Path Problems." *SIAM Journal on Computing* 30, (2001): 809-837.
- Beckman, R. J., K.A. Baggerly, and M.D. McKay. "Creating Synthetic Baseline, Populations." *Transportation Research A* 30(6), (1996): 415-429.
- Ben-Akiva, M. "Network State Estimation and Prediction for Real-Time Traffic Management." *Networks and Spatial Economics* 1, (2001): 293-318.
- DynaMIT, [www.dynamictrafficassignment.org](http://www.dynamictrafficassignment.org), (Accessed in February 2005).
- DYNASMART, [www.dynamictrafficassignment.org](http://www.dynamictrafficassignment.org), (Accessed in February 2005).
- Glover, F., D. Klingman, and N. Phillips. "A New Polynomially Bounded Shortest Paths Algorithm." *Operations Research* 33, (1985): 65-73.
- Hobeika, A.G., M. Jeihani, and C. Jeenanunta. "Transportation Analysis and Simulation Systems (TRANSIMS): Short Course Instructions Manual," Virginia Tech, Blacksburg, 2002.
- Jeihani, M., H.D. Sherali, and A.G. Hobeika. "Computing Dynamic User Equilibria for Large Scale Transportation Networks." *Transportation* 33(6), (2006a): 598-604.
- Jeihani, M., K. Ahn, H.D. Sherali, A.G. Hobeika, and H.A. Rakha. "Comparison of TRANSIMS' Light Duty Vehicle Emissions with On-Road Emission Measurements." *Journal of Transportation Research Forum* 45(1), (2006b): 87-99.

Dynamic Traffic Assignment Computer Packages

Leonard, D.R., J.B. Tough, and P.C. Baguley. "CONTRAM: A Traffic Assignment Model for Predicting Flows and Queues During Peak Periods." *Transportation and ROAD Research Laboratory TRRL Laboratory Report 841*, (1978).

Mahmassani, H.M. "Dynamic Traffic Assignment and Simulation Methodology for Advanced System Management Applications." *Networks and Spatial Economics* 1, (2001): 267-292.

Marcotte, P. and L. Wynter. "A New Look at the Multiclass Network Equilibrium Problem." *Transportation Science* 38(3), (2004): 282-292.

Nagel, K., M. Rickert, P.M. Simon, and M. Pieck. "The Dynamics of Iterated Transportation Simulation." Los Alamos National Laboratory, LA-UR 98-2168, Los Alamos, NM, 1998.

Peeta, S. and A. Ziliaskopoulos. "Foundations of Dynamic Traffic Assignment: The Past, the Present, and the Future." *Networks and Spatial Economics* 1, (2001): 233-265.

Peeta, S. and H.S. Mahmassani. "System Optimal and User Equilibrium Time-Dependent Traffic Assignment in Congested Networks." *Annals of Operations Research* 60, (1995): 81-113.

Planung Transport Verkehr AG (PTV). *VISSIM 3.5 User Manual*, Germany, 2000.

Portland Study, Documentation, <http://transims.tsasa.lanl.gov>, 2001.

Quadstone, Ltd. *Paramics Traffic Simulation Modeler V3.0: Reference Manual*. Quadstone, Edinburgh, UK, 1999.

Raney, B. and K. Nagel. "Iterative Route Planning for Modular Transportation Simulation," *Department of Computer Science, ETH, Zurich, Switzerland*, [www.inf.ethz.ch/personal/nagel/papers](http://www.inf.ethz.ch/personal/nagel/papers), 2002.

Sheffi, Y. *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*. Prentice Hall, Englewood Cliffs, NJ, 1985.

Sherali, H.D. "On the Equivalence Between Some Shortest Path Algorithms." *Operations Research Letters* 10, (1991): 61-65.

Sherali, H.D., C. Jeenanunta, and A. G. Hobeika. "The Approach-Dependent, Time-Dependent, Label-Constrained, Shortest Path Problem." *Networks* 48(2), (2006): 57-67.

Sherali, H.D., A.G. Hobeika, and S. Kangwalklai. "Time-Dependent Label-Constrained Shortest Path Problems with Applications to TRANSIMS: A Transportation Planning Model." *Transportation Science* 37, (2003): 278-293.

Taylor, N.B. "The CONTRAM Dynamic Traffic Assignment Model." *Networks and Spatial Economics* 3, (2003): 297-322.

Wiedemann, R. "Simulation des Strassenverkehrsflusses in Schriftenreihe des Instituts für Verkehrswesen der Universität Karlsruhe." *Heft 8*, 1974.

**Mansoureh Jeihani** received her B.S. in computer engineering from Iran National University (Shahid Beheshti), Tehran, Iran, in 1995, M.S. in socio-economics systems engineering from the Institute for Research in Planning and Development (IRPD), Tehran, Iran, in 1998, M.A. in economics from Virginia Tech, in 2001, and Ph.D. in civil (transportation systems) engineering from Virginia Tech in 2004. She has two years experience as a senior modeling analyst at Resource Systems Group, Inc. and Southern California Association of Governments (SCAG). She is currently an assistant professor at Morgan State University. She is a member of ASCE and ITE.