Journal of Intelligent and Robotic Systems **41:** 173–203, 2004.

Microscopic Traffic Simulation: A Tool for the Design, Analysis and Evaluation of Intelligent Transport Systems

J. BARCELÓ and E. CODINA
Department of Statistics and Operations Research, Universitat Politècnica de Catalunya, Pau Gargallo 5, 08028 Barcelona, Spain; e-mail: jaume.barcelo@upc.es

J. CASAS, J. L. FERRER and D. GARCÍA
TSS-Transport Simulation Systems, Passeig de Gràcia 12, 3°, 1ª, 08007 Barcelona, Spain; e-mail: info@tss-bcn.com

Abstract. This paper summarises some of the main modelling and interface developments made recently in the AIMSUN microscopic traffic simulator to provide a better response to the requirements for the assessment of ITS systems, advanced transport analysis and ATMS. The description addresses two main areas: improvements on the dynamic assignment capabilities, and the embedding of the simulator in the AIMSUN/ISM (Intermodal Strategy Manager) a versatile graphic environment for model manipulation and simulation based traffic analysis and evaluation of advanced traffic management strategies. AIMSUN/ISM includes two specific tools, the Scenario Analysis Module to generate and simulate the traffic management strategies, and the (ODTool) to generate and manipulate the Origin-Destination matrices describing the mobility patterns required by the dynamic analysis of traffic conditions. The matrix calculation procedures have been implemented on basis to a flexible interface with the EMME/2 transport planning software.

Key words: traffic simulation, traffic management, intelligent transport systems.

1. Introduction

Microscopic traffic simulators are very likely the most powerful and versatile traffic analysis tools. Its ability to reproduce to a significant level of accuracy the observed traffic conditions in a broad variety of circumstances makes that the skilled users become very demanding, asking for new features and functionalities in the never ending process of fitting better the increasing complexity of traffic phenomena. This demand for new improvements namely reaches its highest point when ITS applications are involved. Consequently AIMSUN is a continuously evolving traffic simulator, as well as GETRAM, its supporting software modelling platform. AIMSUN (**A**dvanced **I**nteractive **M**icroscopic **S**imulator for **U**rban and Non-Urban **N**etworks; `http://www.aimsun.com`) is imbedded in GETRAM (**G**eneric **E**nvironment for **TR**affic **A**nalysis and **M**odeling), a simulation environment inspired by modern trends in the design of graphical user interfaces adapted to traffic modeling requirements. Among the many features and functions added recently to GETRAM/AIMSUN, (see (TSS, 2002)), we would like to highlight two in this paper: Improvements on the dynamic assignment abilities, and the embedding of the AIMSUN simulator in the AIMSUN/ISM, including a module for the analysis of traffic scenarios.

Microscopic traffic simulators are simulation tools that realistically emulate the flow of individual vehicles through a road network. Most of the currently existing microscopic traffic simulators are based on the family of car-following, lane changing and gap acceptance models to model the vehicle's behavior, for a comprehensive description of car-following models see (Gabbard, 1991). They are proven tools for aiding transportation feasibility studies. This is not only due to their ability to capture the full dynamics of time dependent traffic phenomena, but also because they are capable of using behavioral models that can account for drivers' reactions when exposed to Intelligent Transport Systems (ITS). GETRAM, presented in this paper, is a simulation environment with a microscopic traffic simulator, AIMSUN, at its heart. It provides:

- The ability to accurately represent any road network geometry: An easy to use Graphic User Interface (TEDI) that can use existing digital maps of the road network allows the user to model any type of traffic facility.
- Detailed modeling of the behavior of individual vehicles. This is achieved by employing sophisticated and proven car following and lane changing models that take into account both global and local phenomena that can influence each vehicle's behavior.
- An explicit reproduction of traffic control plans: pre-timed as well as those defined by TRANSYT, SYNCHRO or Nema's standards. Auxiliary interfacing tools that allow the simulator to work with almost any type of real-time or adaptive signal control systems, as C-Regelaar, Balance, SCATS and UTOPIA, are also provided.
- Animated 2D and 3D output of the simulation runs. This is not only a highly desirable feature but can also aid the analysis and understanding of the operation of the system being studied and can be a powerful way to gain widespread acceptance of complex strategies.

Car-following and lane changing models in AIMSUN have evolved respectively from the seminal Gipps models (Gipps, 1981, 1986). The way in which the the car-following model has been implemented in AIMSUN takes into account the additional constraints on the breaking capabilities of the vehicles, imposed in the classical safe to stop distance hypothesis, as in the analysis carried out by Mahut (2000). The implementation tries also to capture the empirical evidence that driver behaviour depends also on local circumstances (i.e. acceptance of speed limits on road sections, influence of grades, friction with drivers in adjacent lanes, and so on). This is done in AIMSUN by means of model parameters whose values, calculated at each simulation step, depend on the current circumstances and conditions at each part of the road network.

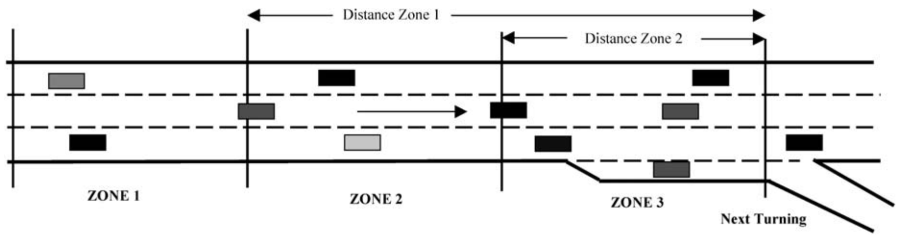


Figure 1. Lane changing zones.

The lane changing process, which also evolves from Gipps' model, is modelled as a decision process that emulates the driver's behaviour considering the need to change lane (as in the case of turning manoeuvres determined by the route), the desirability of the lane change (as, for example, to overpass a slow moving vehicle), and the feasibility conditions for the lane change. Lane change also depends on the location of the vehicle on the road network. To achieve a more accurate representation of the driver's behavior in the lane changing decision process, three different zones inside a section are considered, each one corresponding to a different lane changing motivation. The distance up to the end of the section characterizes these zones and the next turning point. Figure 1 depicts the structure of these zones that are defined as follows:

- Zone 1: This is the farthest from the next turning point. The lane changing decisions are mainly governed by the traffic conditions of the lanes involved.
- Zone 2: This is the intermediate zone. In this zone vehicles look for a gap and may try to accept it without affecting the behavior of vehicles in the adjacent lanes.
- Zone 3: This is the nearest to the next turning point. Vehicles are forced to reach their desired turning lanes, reducing the speed if necessary and even coming to a complete stop in order to make the lane change possible. Also, vehicles in the adjacent lane can modify their behavior in order to allow a gap big enough for the lane-changing vehicle.

The length of the lane changing zones is defined by two parameters, distance zone 1 and distance zone 2, whose values depend on the current traffic conditions on the road section at each simulation step. For a more detailed description see (Barceló and Casas, 2002; Barceló, 2001) or (TSS, 2002).

Vehicles are assigned to routes according to a route choice model (Barceló et al., 1995). Additionally AIMSUN allows vehicles to change their chosen route from origin to destination according to variations in traffic conditions as they travel through the road network. This provides the basis for heuristic traffic assignment.

The recent evolution of the AIMSUN microscopic simulator has taken advantages of the state-of-the-art in the development of object-oriented simulators, and graphical user interfaces, as well as the new trends in software design and the available tools that support it adapted to traffic modelling requirements (Banks, 1998).

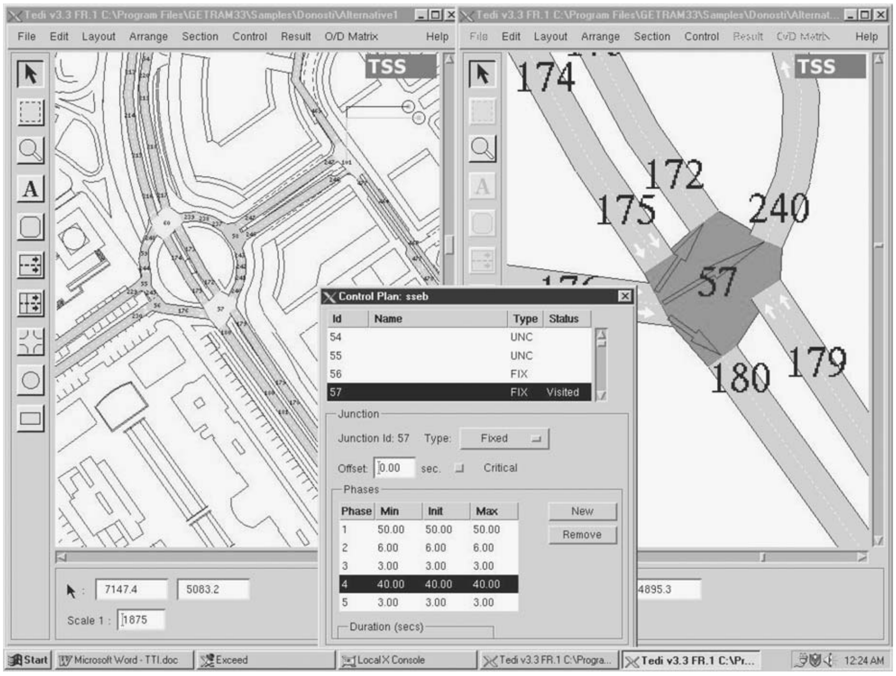


Figure 2. Example of GETRAM graphic user interface for building microscopic simulation models.

A proper achievement of the basic requirements of a microscopic simulator implies building models as close to reality as possible. The closer the model is to reality the more data demanding it becomes. This has been traditionally the main barrier preventing wider use of microscopic simulation. Manual coding of geometric data, turning movements at intersections, timings and so on, is not only cumbersome and time consuming but also a potential source of errors. It is also hard to debug if the appropriate tools are not available. A way of overcoming these drawbacks has been to provide GETRAM/AIMSUN with the proper user friendliness based on the versatility of the TEDI traffic network graphical editor, which can import the geometric background of the road network to draw the network model on top, as shown on the left part of Figure 2.

The background can be imported as a .dxf file from a CAD or GIS system, or any other graphic format as .jpg, bit map and so on. All objects comprising the road model can be built with the graphic editor. Their attributes and parameters are defined and assigned values by means of window dialogues such as the one in the right part of Figure 2, which shows the definition of the shared movements in a phase of a pre-timed signal control, and the allocation of the timings. Summarizing, this software environment for traffic modeling makes an easy task of the model building process, ensures accurate geometry, prevents errors, provides powerful debugging tools and can model any type of traffic related facility.

2. Heuristic Dynamic Assignment

Advanced driver information systems, adaptive traffic control systems, real-time traffic management systems, and so on, are examples of the so-called Intelligent Transport Systems currently under development or, in some cases, already being tested in field trials. Common requirements for the analysis of these systems are: Practical methods of measuring the degree of change in activity flows resulting from system modifications, real-time identification of imbalance situations in the use of available capacity of the road network, definition and assessment of suitable strategies, and implementation of real-time management decisions and control measures.

The development of systems successfully fulfilling these conditions requires traffic models that efficiently represent interactions in the transport system dynamically. Therefore, they should take proper account of the effects of time-varying demand, time dependent queuing and so on. Microscopic traffic simulation has proven its usefulness in many different areas of application dealing with complex traffic systems (Barceló and Casas, 1999; Barceló et al., 1999a, 1999b, 1999c; Barceló, 2000). Simulation has consequently been proposed as numerical procedure, heuristic in nature, able of providing approximate solutions to dynamic traffic systems, not only due to its ability to capture the full dynamics of time dependent traffic phenomena, but also for being capable of dealing with behavioral models accounting for drivers' reactions when exposed to Intelligent Transport Systems (ITS). To achieve these objectives a microscopic simulator should be able of: Updating timely the routes from origins to destinations depending on changing conditions of traffic over time, assigning the vehicles to routes from origins to destinations at each time period, and dynamically re-route part of the vehicles en-route when better alternative routes from their current position to their destination exist.

This type of simulation assumes that the demand is defined in terms of origin–destination matrices whose entries represent the number of trips from an origin to a destination as a function of time (Barceló et al., 1995). The routes are calculated according to specified travel costs and the assignment to the routes is based on modeling driver's decisions by means of route choice models. The heuristic dynamic assignment procedure works as follows:

1. Calculate initial shortest routes, taking the estimated initial costs.
2. Simulate for a period using available route information and obtain new costs as a result of the simulation.
3. Recalculate shortest routes, taking into account the new costs.
4. Add the new information calculated in 3 to the knowledge of the drivers.
5. Go to step 2.

At the beginning of the simulation, shortest path trees are calculated from every section to each destination centroid, taking as arc costs the specified initial costs. During simulation, new routes are recalculated in every time interval, taking the specified arc costs, that have been updated for each arc once the last interval statistics have been gathered. For each destination and time period, the optional routes are stored as a tree that allows knowing how to reach the destination from any section of the network. The shortest route component takes into account turning penalties, as the different turning movements at the end of a section have in general unequal travel times (e.g., left turn, drive through, etc.). The procedure implemented to compute the shortest routes to a destination uses a network where an arc, connecting two nodes, models a section. A special arc connecting the beginning of the turning to its end models a turning movement. The computation of shortest routes uses a label setting method, where the labels are associated with an arc. The network is constructed only once, just before the start of the simulation. During the simulation, the computation of shortest routes is launched at certain time steps. The shortest route routine is a variation of Dijkstra's label setting algorithm. It provides the shortest routes from the start of every section to all destinations. The penalties associated with turning movements are taken into account.

The experience and computational tests have shown the importance of accounting for the users' perception when defining the available routes. To accomplish this objective the new version of AIMSUN offers three possibilities:

(a) Historical Fixed Routes (HFR): Predefined fixed routes, set manually by means of the network editor or imported from the output of other traffic simulators (macroscopic, i.e. EMME/2, or microscopic);
(b) Historical Shortest Path Trees (HSPT): Predefined shortest paths, which can be imported from the output of previous simulations with AIMSUN or another traffic simulators, and
(c) Calculated Shortest Path Tree (CSPT): Shortest path tree calculated using the cost functions. (There are two types of CSPT: Initial Shortest Path Tree (ISPT): for each destination centroid, it gives a shortest path tree, using the initial cost function for each turning movement, and Computed Shortest Path Tree (SSPT): shortest path trees computed at each time period for each destination centroid, using cost functions that depend on the statistical data gathered during the simulation.)

A vehicle with vehicle type v_t traveling from origin O_i to destination D_j, could choose the route among the following possible paths: The N predefined Historical Fixed Routes: $\mathrm{HFR}_k(\mathrm{O}_i, \mathrm{D}_j),\ k = 1, \ldots, N$, the M predefined Historical Shortest Path Trees: $\mathrm{HSPT}_k(\mathrm{D}_j),\ k = 1, \ldots, M$, the 1 Initial Shortest Path Trees at the beginning of the simulation: $\mathrm{ISPT}(\mathrm{D}_j)$, and the P Computed Shortest Path Trees: $\mathrm{SSPT}_k(\mathrm{D}_j),\ k = 1, \ldots, P$.

The user may define the time interval for recalculation of the computed shortest paths and the maximum number of path trees to be maintained during the simulation. When the maximum number of path trees (K) is reached, the oldest paths will be removed as soon as no vehicle is following them. It is assumed that vehicles only choose between the most recent K path trees. Therefore, the oldest ones will become obsolete and disused. From the point of view of the practitioner

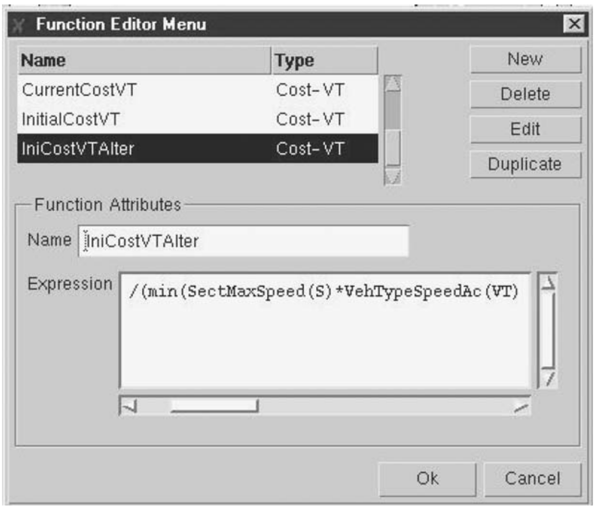


Figure 3. Example of User defined arc cost function per vehicle type.

the answers provided to two main questions could heavily condition the use of the heuristic dynamic assignment as analysis tool: the concept of cost used in updating the routes, and the route choice model used in assigning vehicles to available routes.

Assuming that route cost is the sum of the costs of the arcs composing the route, a wide variety of arc costs can be proposed: travel times at each simulation interval, toll pricing, historical travel times representing driver's experience from previous days, combinations of various arc attributes as for instance travel times, length and capacity, etc. The improved version of AIMSUN provides the user with two alternatives: use the default initial and cost functions to calculate the arc costs or use the Function Editor included in TEDI (Network Editor in the GETRAM modeling environment) to define his/her own arc cost function using any of the most common mathematical functions and operators, and as arguments any of the numerical attributes, statistical values or vehicle characteristics available in the simulator. Calculation of shortest paths is carried out per vehicle type, taking into account reserved lanes. Therefore, the set of paths from which a vehicle will select one, either when entering the network or when being re-routed, may be different for different vehicle types traveling to the same destination depending on the presence of elements like reserved lanes or tolls. Also the travel time used in the cost function for recalculation of shortest paths is taken as the travel time per vehicle type. Figure 3 illustrates an example of a simulation model for which alternative arc cost functions have been defined by the user. The open window shows part of the algebraic expression for a cost function per vehicle type.

In a similar way when using these dynamic assignment abilities it could be raised the question of which is the most suitable route choice function. Route choice functions represent implicitly a model of user behaviour, representing the

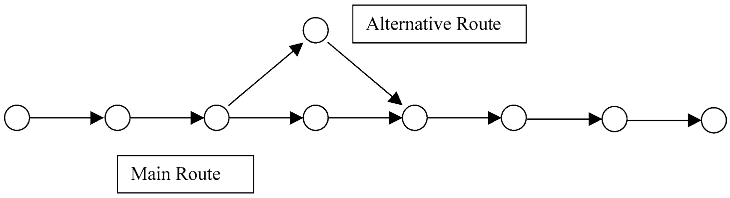


Figure 4. Overlapping routes.

most likely criteria employed by the user to decide between alternative routes: perceived travel times, route length, expected traffic conditions along the route, etc. The solution implemented in the most recent version of AIMSUN also provides the user two alternatives: use the default functions or define his/her own route choice function by means of the Function Editor. The most used route choice functions in transportation analysis are those based on the discrete choice theory, i.e. Logit functions assigning a probability of a route being chosen to each alternative route between each origin-destination pair depending on the difference between the perceived utilities.

A drawback reported in using the Logit function is the exhibited tendency towards route oscillations in the routes used, with the corresponding instability creating a kind of flip-flop process. According to our experience there are two main reasons for this behavior. The properties of the Logit function and the inability of the Logit function to distinguish between two alternative routes when there is a high degree of overlapping. When the network topology allows for alternative routes with little or no overlapping at all, the instability of the routes used can be substantially improved playing with the shape factor of the Logit function and re-computing the routes very frequently. However, in large networks where many alternative routes between origin and destinations exist and some of them exhibit a certain degree of overlapping (see Figure 4), the use of the Logit function may still exhibit some weaknesses (Ben-Akiva and Bierlaire, 1999; Cascetta et al., 1996).

To avoid this drawback the C-Logit model (a variation of the Logit model introduced by Cascetta et al. (1996)) has been implemented. In this model, the choice probability P_k, of each alternative path k belonging to the set I_{rs} of available paths connecting an O/D pair, is expressed as:

$$P_k = \frac{\mathrm{e}^{\theta(V_k - \mathrm{CF}_k)}}{\sum_{l \in I_{\mathrm{rs}}} \mathrm{e}^{\theta(V_l - \mathrm{CF}_l)}}, \tag{1}$$

where V_l is the perceived utility for alternative path l, and θ is the scale factor, as in the case of the Logit model. The term CF_k, denoted as 'commonality factor' of path k, is directly proportional to the degree of overlapping of path k with other alternative paths. Thus, highly overlapped paths have a larger CF factor and therefore smaller utility with respect to similar paths. CF_k is calculated as follows:

$$\mathrm{CF}_k = \beta \cdot \ln \sum_{l \in I_{\mathrm{rs}}} \left(\frac{L_{lk}}{L_l^{1/2} L_k^{1/2}} \right)^{\gamma}, \tag{2}$$

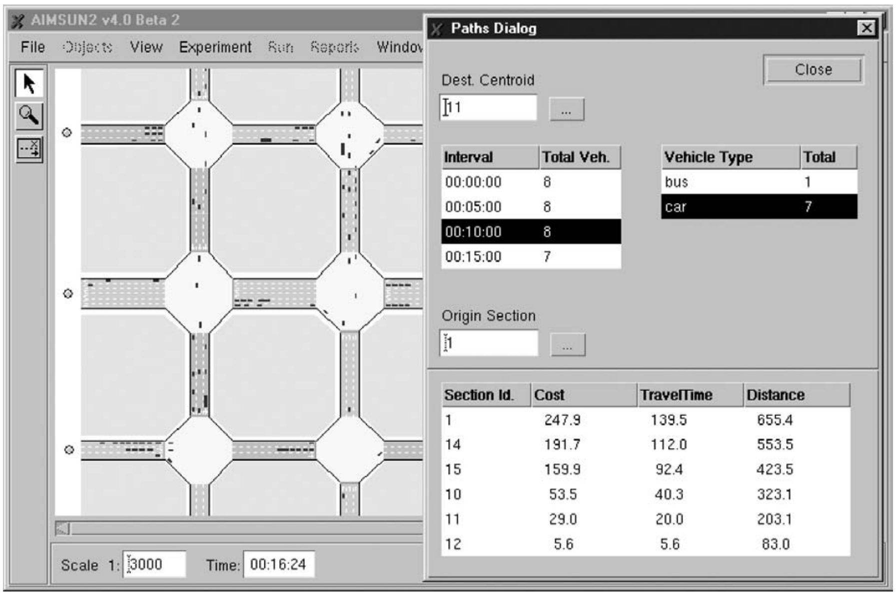


Figure 5. Path dialog window.

where L_{lk} is the length of arcs common to paths l and k, while L_l and L_k are the length of paths l and k, respectively. Depending on the two factor parameters β and γ, a greater or lesser weighting is given to the 'commonality factor'. Larger values of β means that the overlapping factor has greater importance with respect to the utility V_l; γ is a positive parameter, whose influence is smaller than β and which has the opposite effect. The utilities considered in traffic simulation are the opposite of the arc costs, as correspond to negative utilities.

To get the insight on what is happening in a heuristic dynamic assignment, for the proper calibration and validation of the simulation model, the user should have access to the analysis of the used routes. To support the user in this analysis process AIMSUN includes a path analysis tool. Figure 5 depicts the path dialogue window. The path list box contains the list of section identifiers composing the path and the following information is displayed for each section: the cost, according to the arc cost function used, the current arc travel time in seconds, and the length of the path. The information is provided for each time interval at which costs and paths are updated according to the changes in traffic condions. The arc costs, and therefore the paths costs, are in this way time dependent, and change dynamically along the simulation.

In the example shown in Figure 5, at the five minute time interval from 00:10:00 until 00:15:00, the shortest path from section 1 to centroid 11 goes through sections 14, 15, 10, 11 and 12. The cost of the whole path is 247.9 units (depending on the definition of cost), the travel time is 139.5 seconds and the distance is 655.4 meters. In this case cost and travel time are different.

Given an OD pair (r, s) with origin O_r and destination D_s, if P_{rs} is the set of feasible paths from O_r to D_s and I_T is the set of time intervals to account for, the path analysis tool makes available to the user information on: the current travel time on path k at time interval t, where $k \in P_{rs}$ and $t \in I_T$; the historical travel time on path k at time interval t, where $k \in P_{rs}$ and $t \in I_T$; the current flow on the arcs along the path at time interval t, where $k \in P_{rs}$ and $t \in I_T$, the saturation index on path k at time interval t. This information allows to have: time plots of path travel times, time plots of path saturation indexes, and makes also possible the calculation of utilities associated to all paths for the analysis of day-to-day and within-day traffic variations.

From the behavioral point of view many analysts propose a route choice mechanism based on a day-to-day learning mechanism, assuming that user's choice is based on a perception of the experienced past path costs. To emulate this process the simulation is replicated N times, and link costs for each time interval and every replication are stored. Thus at iteration l of replication j the costs for the remaining $l+1, l+2, \ldots, L$ (where $L = T/\Delta t$, T being the simulation horizon and Δt the user defined time interval to update paths and path flows) time intervals for the previous $j-1$ replications can be used in an anticipatory day-to-day learning mechanism to estimate the expected link cost at the current iteration. Let $s_a^{jl}(v)$ be the current cost of link a at iteration l of replication j, then the average link costs for the future $L-l$ time intervals, based on the experienced link costs for the previous $j-1$ replications is

$$\bar{s}_a^{j,l+i}(v) = \frac{1}{j-1}\sum_{m=1}^{j-1} s_a^{m,l+i}(v), \quad i = 1, \ldots, L-l. \tag{3}$$

The "forecasted" link cost can then be computed as

$$\tilde{s}_a^{j,l+1}(v) = \sum_{i=0}^{L-l} \alpha_i \bar{s}_a^{j,l+i}(v),$$
$$\text{where} \quad \sum_{i=0}^{L-l} \alpha_i = 1, \quad \alpha_i \geqslant 0,\ \forall i,\ \text{are weighting factors.} \tag{4}$$

The resulting cost of path k for the ith OD pair is

$$\tilde{S}_k(h^{l+1}) = \sum_{a \in A} \tilde{s}_a^{j,l+1}(v)\delta_{ak} \tag{5}$$

where, as usually, δ_{ak} is 1 if link a belongs to path k and 0 otherwise. The path costs $\tilde{S}_k(h^{l+1})$ are the arguments of the route choice function (logit, C-logit, user defined, etc.) used at iteration $l+1$ to split the demand g_i^{l+1} among the available paths for OD pair i.

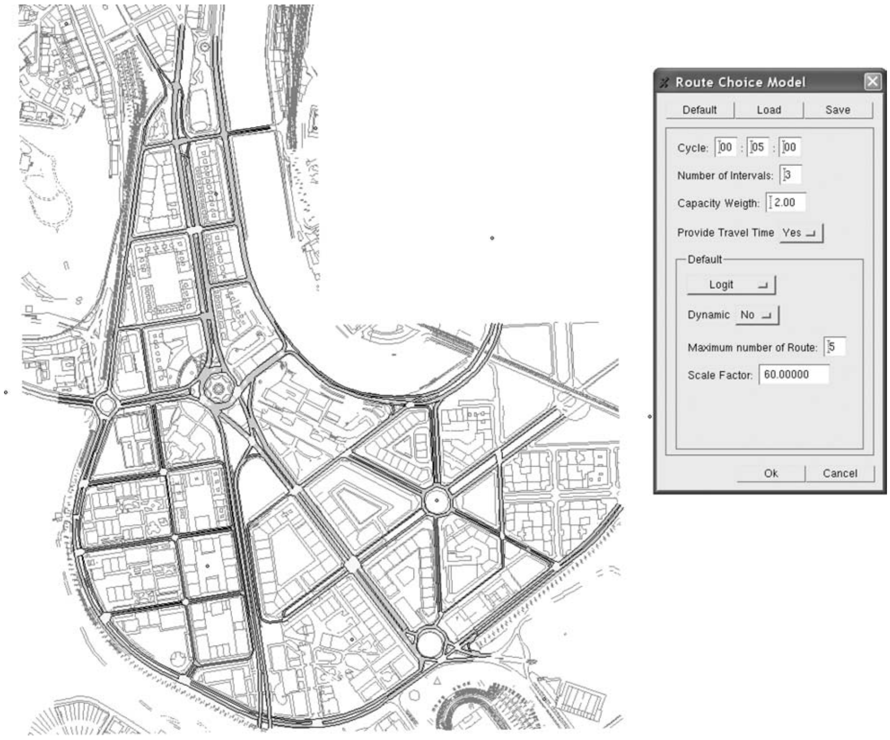


Figure 6. Simulation model of Amara and definition of the route-based simulation.

2.1. AN EXAMPLE OF ROUTE BASED SIMULATION: HEURISTIC DYNAMIC ASSIGNMENT IN A URBAN ENVIRONMENT

Figure 6 depicts the AIMSUN simulation model of the Amara borough in the city of San Sebastian in Spain. The window dialogue on the right upper corner of the figure shows the dialogue to select the route choice model to be used in the simulation experiment. In this case arc costs and paths will be updated every 5 minutes using data from 3 previous time intervals. Arc costs will include penalties on section capacities to avoid the selection of strange paths. The route choice function selected, to estimate the probability of a vehicle being assigned to an available path is the Logit funtion, with a shape parameter of 60. A maximum of 3 alternative routes for each origin–destination will be taken into account in the experiment.

To illustrate how works the heuristic assignment described in the previous sections, two alternative paths A, and B from origin 7 (North) to destination 2 (South) have been selected. Figure 7 depicts the selected paths.

Trips from origin 7 to destination 2 are distributed between Paths A and B during the simulation according to the varying traffic conditions which will determine the time dependent path costs used by the logit route choice function. The graphic in Figure 8 describes the time evolution of the percentual trip distribution for a two hours simulation. A critical question when using simulation concerns the validity

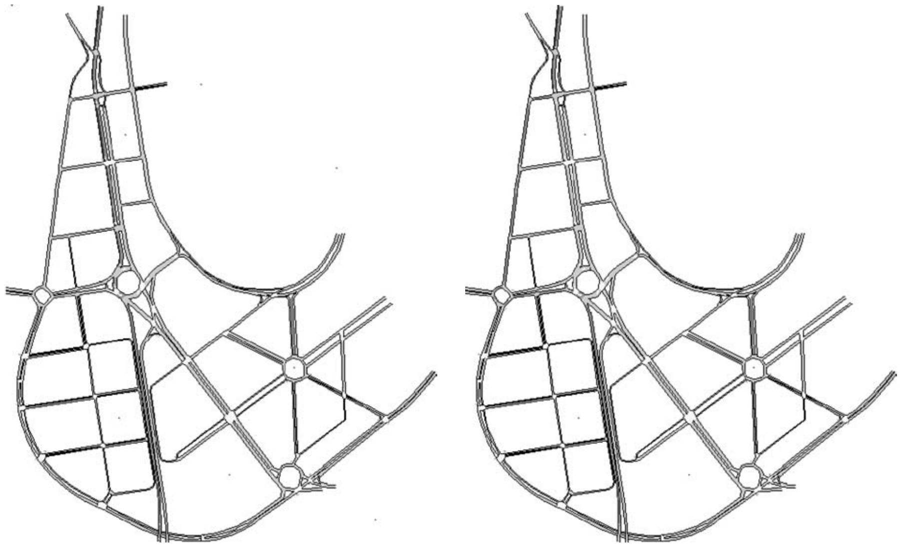


Figure 7. Paths A from origin 7 (North) to destination 2 (South) and B from origin 7 (North) to destination 2 (South).

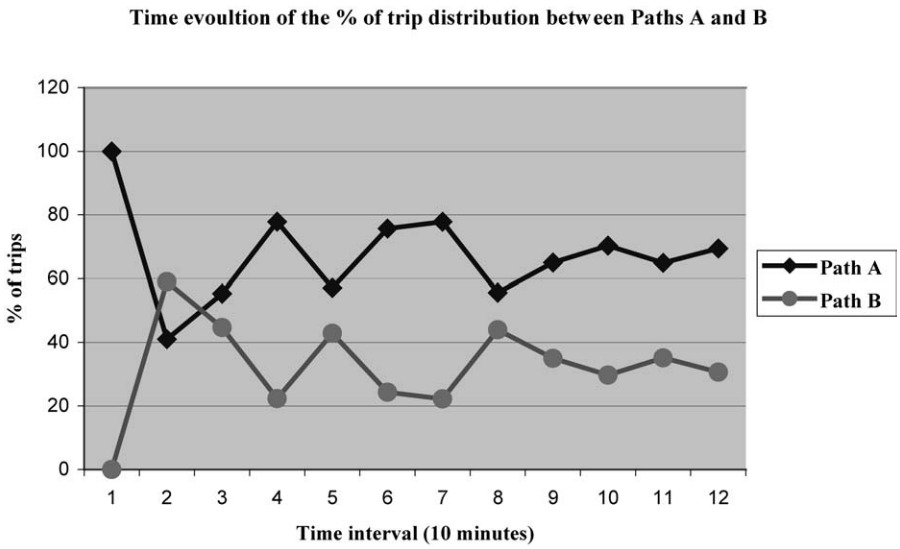


Figure 8. Evolution of the percentage of trip distribution between alternative paths A and B.

of the model, that is the correspondence between the observed reality and the simulation model representing it. In the case of the traffic simulation the validation of the model is established in terms of the comparison at specific points in the network (i.e. where the traffic detectors are located) between the observed values for the traffic variables (i.e. traffic flows, average speeds, or occupancies) and the simulated values for the same traffic values at the same points in the model (for

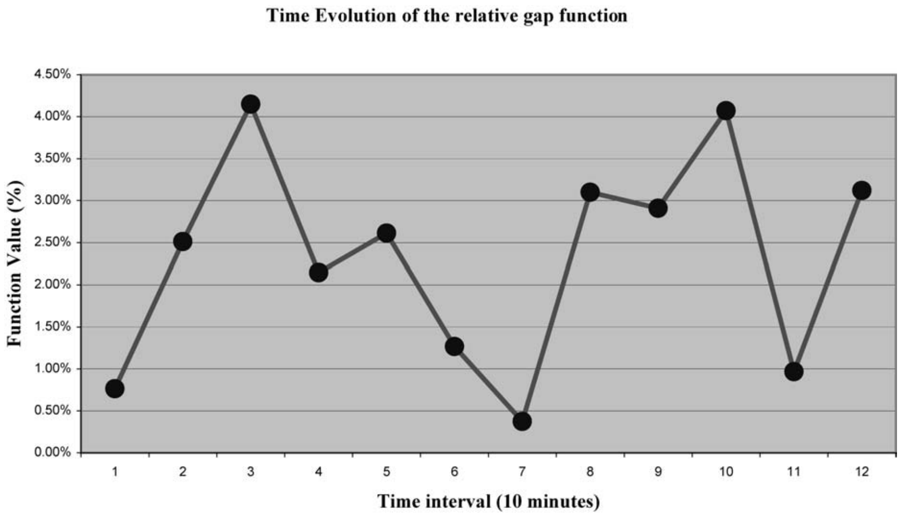


Figure 9. Time evolution for the relative gap function for the simulation experiment with the Amara model.

a detailed discussion see (Barceló and Casas, 2002). However, from the point of view of the dynamic assignment an alternative way of considering the quality of the model can be established in terms of the relative gap function (Florian et al., 2001), a function that measures how close is the use of the alternative paths to those expected from a user equilibrium, in which travel times along the used paths should be equal.

Defining the relative gap function at time interval t as

$$RG(t) = \frac{\sum_{i\in I}\sum_{k\in K_i} h_k(t)[s_k(t) - u_i(t)]}{\sum_{i\in I} g_i(t)u_i(t)}, \tag{6}$$

where $h_k(t)$ is the flow on path k, and $s_k(t)$ is the cost of path k at time interval t, $g_i(t)$ is the traffic demand for the ith OD pair at time interval t, $u_i(t)$ is the cost of the minimum cost path for the ith OD pair, K_i is the set of all alternative paths for the ith OD pair, and I is the set of all OD pairs. The graphic in Figure 9 shows the time evolution of the relative gap function for the defined simulation experiment with the Amara model. As it can be seen the relative gap oscillates between 0.5% and 4.00%, which is quite acceptable, and correeponds to the real oscillation between routes to the same destination.

3. AIMSUN/ISM: the Scenario Analysis Module

The *Scenario Analysis Module* in AIMSUN/ISM uses an AIMSUN microscopic traffic simulation model of the traffic network under study to define, verify and optimise traffic management strategies, evaluate the expected impacts of the strategies and determine the triggers to activate strategies according to prevailing traffic conditions. The *Scenario Analysis Module* in AIMSUN/ISM is a new graphic software platform that embeds the microscopic traffic simulator AIMSUN, and interfaces the transport planning software EMME/2, see (INRO, 1999), providing the analyst with a friendly user interface to perform the above described operations.

A scenario is a microscopic simulation model of a traffic network, or a subnetwork of a large network, in which a traffic problem has been identified: the so-called problem network. The model input reproduces to a great degree of accuracy the traffic demand in the problem network for the time period for which the traffic problem has been identified, as well as the current operational conditions in the road network (i.e. current traffic control at signalized intersections, reductions of capacity at specific parts of the network by road works, incidents, and so on). The analysis of the scenario consists on a set of simulation experiments whose purpose is to help the traffic manager to develop and evaluate the impacts of the single actions or combination of actions, consisting of situation related measures (i.e. reroutings and/or speed control using Variable message Panels (VMS), changes in control, an so on), with the objective of alleviating or eliminating the traffic problem identified. This concept of action composed by the various situation-related measures is called a strategy. The evaluation of alternative scenarios, i.e. models of the same problem network with alternative traffic management strategies, is based on the comparison of the values of performance indexes measuring saturation levels, quality of service, total travel time, average delays, average queue lengths or total vehicle-kilometers traveled.

The *Scenario Analysis Module* in AIMSUN/ISM (Barceló et al., 2002b), is based on a combination of an AIMSUN microscopic traffic simulation model and an EMME/2 transport planning model of the traffic network, providing the analyst with the tools for building the specific scenarios for the subnetworks of the road network where the traffic problems have been identified, and defining, verifying and optimising traffic management strategies, evaluate the expected impacts of the strategies and determine the triggers, that is the threshold values of the traffic variables (volumes, occupancies or speeds at specific part of the road network) to activate strategies according to prevailing traffic conditions. Three auxiliary tools assist AIMSUN/ISM operation:

- GETRAM/TEDI: The Generic Environment for Traffic Analysis and Modelling and its associated graphic editor TEDI that supports the network edition.
- AIMSUN, the microscopic traffic simulator providing the dynamic traffic models for the evaluation of the traffic management strategies, interactively activated from AIMSUN/ISM.
- EMME/2 a transport planning software providing the macroscopic traffic models for traffic assignment and O/D matrix adjustment to deal with the analysis of the demand patterns for the selected scenarios.

The main objective of AIMSUN/ISM is to allow the fast and convenient manipulation of input data to create simulation scenarios and to present result data

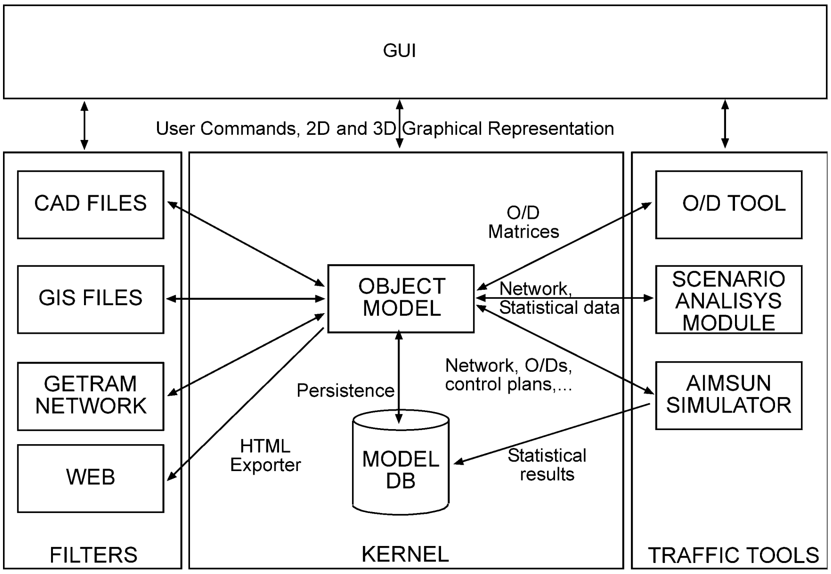


Figure 10. AIMSUN/ISM functional architecture.

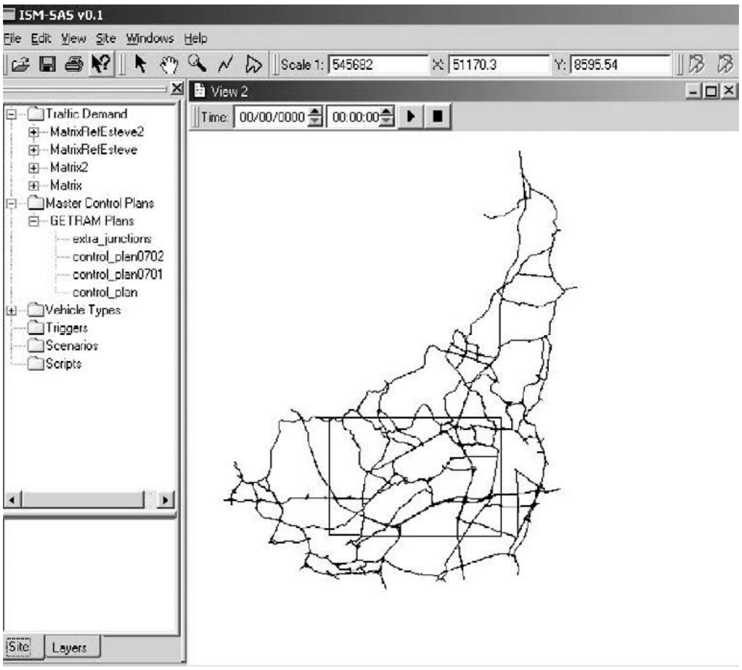
in a compressible way. It has two main components: The simulation experiment specification and the result analysis. The simulation experiment specification includes: The set-up of a problem network (either the network of the whole area or a subnetwork); the creation, modification and adjustment of O/D matrices (global for the whole area as well as local or traversal for the subnetworks); the addition of traffic management policies and their triggers and the simulator tuning.

The result analysis includes: the output data presentation and the comparative study of the performance of a solution, either with previous solutions or with real data.

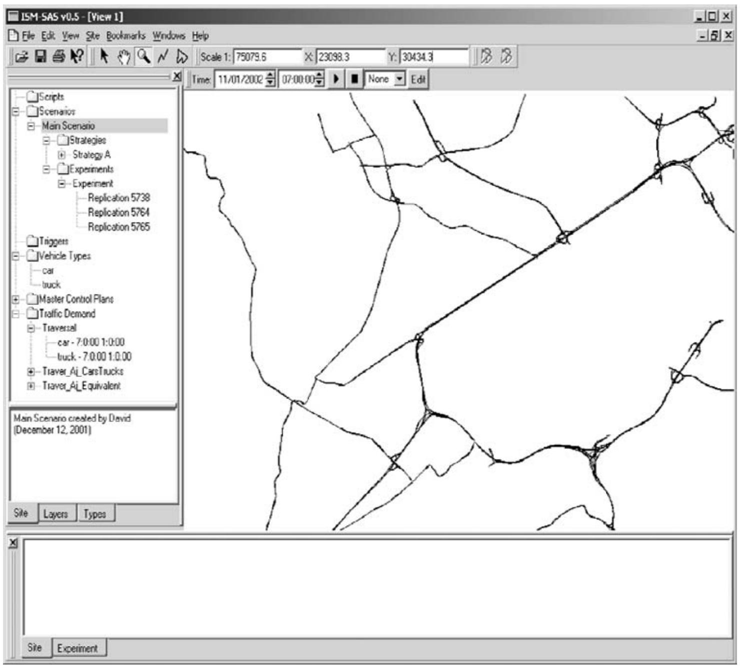
The high level diagram of the functional architecture of AIMSUN/ISM is depicted in Figure 10.

Since a problem can have different solutions and since these solutions cannot be obvious, the user can define several experiments combining different policies until he/she finds the best option. During this experimentation the user can reuse previous solutions and add new ones. Then the user can compare the performance of the new solution with either real data or other solutions. These two components can be used iteratively until a satisfactory solution is found. These components provide the support for the generation, evaluation and optimisation of traffic management strategies.

The AIMSUN/ISM operation is illustrated in Figure 11. In this figure global network, and a potential problem network are shown. A problem network corresponds to a subnetwork of the road network on which a specific traffic problem may arise or is identified by the user. The user, by opening a window on the screen on which the WAYFLOW network is displayed, defines the problem network graphically.


ISM-SAS v0.1
File Edit View Site Windows Help
Scale 1: 545682
X: 51170.3
Y: 8595.54
View 2
Time: 00/00/0000
00:00:00
Traffic Demand
MatrixRefEsteve2
MatrixRefEsteve
Matrix2
Matrix
Master Control Plans
GETRAM Plans
extra_junctions
control_plan0702
control_plan0701
control_plan
Vehicle Types
Triggers
Scenarios
Scripts
Site
Layers

(a)


ISM-SAS v0.5 - [View 1]
File Edit View Site Bookmarks Windows Help
Scale 1: 75079.6
X: 23098.3
Y: 30434.3
Time: 11/01/2002
07:00:00
None
Edit
Scripts
Scenarios
Main Scenario
Strategies
Strategy A
Experiments
Experiment
Replication 5738
Replication 5764
Replication 5765
Triggers
Vehicle Types
car
truck
Master Control Plans
Traffic Demand
Traversal
car - 7:0:00 1:0:00
truck - 7:0:00 1:0:00
Traver_Aj_CarsTrucks
Traver_Aj_Equivalent
Main Scenario created by David
(December 12, 2001)
Site
Layers
Types
Site
Experiment

(b)

Figure 11.

The rectangle in Figure 11(a) corresponds to the selected problem network. A problem network is characterized by: the road network within the defining window, and *n* OD database linked to the problem network with the various demand patterns for the problem network under various circumstances (season, day of the week, time of the day, special event, etc.).

A strategy database containing the specifications of the potential traffic management strategies to operate on the Problem Network depending on the identified or potential traffic problem and the demand pattern. The operation of the site creation and problem network definition in AIMSUN/ISM is also illustrated in Figure 11(a), where the AIMSUN/ISM working area displays the model of the road network of the site (the WAYFLOW road network of Hessen in this case). The rectangle drawn by the operator on the network model corresponds to the problem network. Once the problem network has been defined the operator activates the extraction of the subnetwork model for the problem network, which is the first step in the process of generating the scenario to be analyzed and also the automatic production of the AIMSUN microscopic simulation model to use on the analysis of the scenario and on the assessment of the potential impact of the proposed traffic management strategies intended to alleviate the identified traffic problem. Figure 11(b) depicts the automatically generated GETRAM model for the problem network.

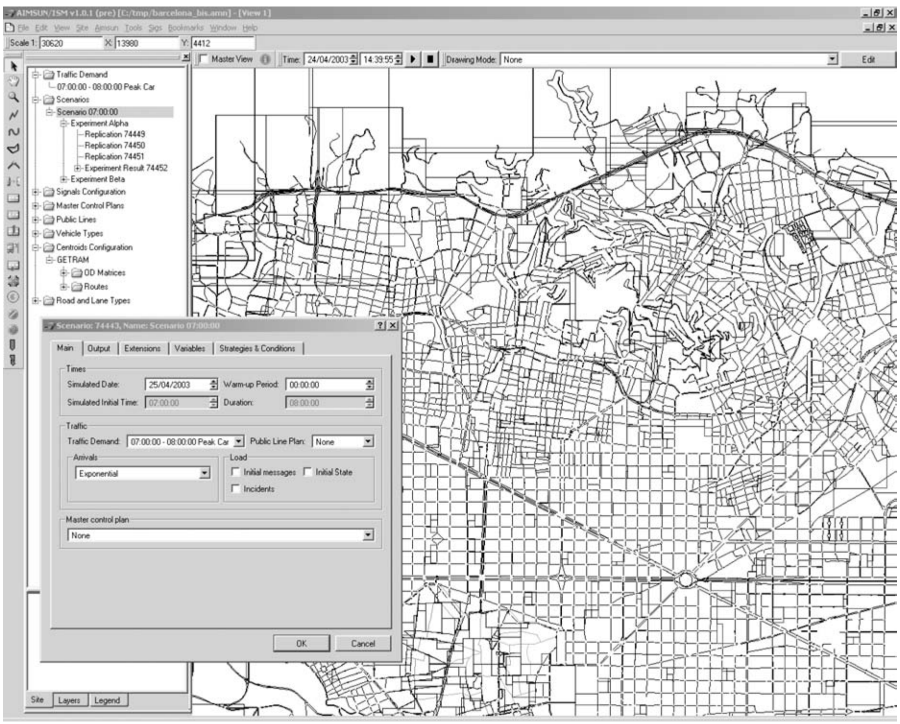


Figure 12. Scenario Analysis Module in AIMSUN/ISM: GUI Dialogue to generate a scenario (I).

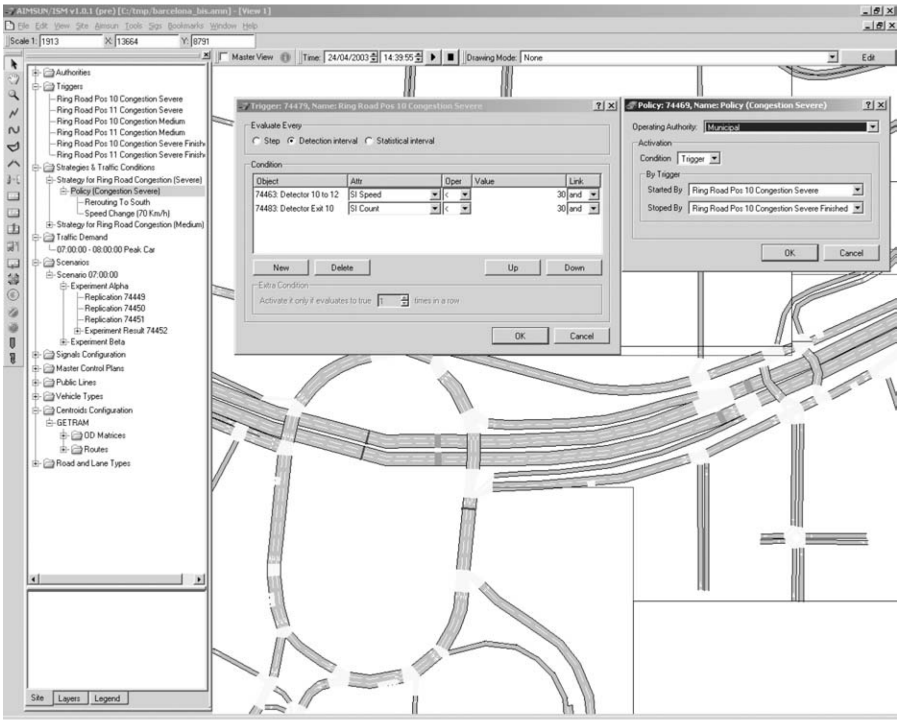


Figure 13. Scenario Analysis Module in AIMSUN/ISM: GUI Dialogue to generate a scenario (II).

Once the GETRAM model has been created in the working area of the Scenario Analysis Module in AIMSUN/ISM, the specification of the scenario is completed by defining the input data (traffic demand) and all the complementary information required to run the simulation experiments. Figures 12 and 13 depict examples of such dialog as supported by the Graphic User Interface (GUI) of the Scenario Analysis Module. In the dialogue shown in Figure 12 the dialogue to define the input data for a specific scenario are depicted in the associated window. The dialogue in Figure 13 corresponds to the definition of a strategy and the conditions that will activate the strategy. It is defined in terms of logical conditions related to the triggers, as for example: if the speed on a specific detector or set of detectors drops below 30 Km/h (trigger) then activate the set of actions composing the strategy called “Ring Road Pos 10 Congestion Severe” (this example corresponds to a study done for the Ring Roads of the city of Barcelona).

4. O/D Calculation (OD Tool)

AIMSUN/ISM assumes explicitly that the traffic analysis tools that it contains, and namely the microscopic simulation with AIMSUN, provide the support for a dynamic analysis of traffic scenarios that takes into account the time variability of traffic phenomena. That means that the analysis tools require inputs describing the traffic mobility patterns and, if possible, their time dependencies. For example, the proper assessment of the impacts of management strategies implying rerouting and diversion needs such type of input. A way of providing this input is through the appropriated OD matrices. The objective of the OD tool is to provide a module supporting the functions that can generate the requested input. Examples of such functions are: matrix edition, generation of the local traversal OD matrix for the selected problem network and time period, adjustment of the local traversal from the available traffic counts for that time period in order to account for the explicit time dependencies, Modification of the adjusted traversal to account for increases or decreases in the traffic demand at given zones to deal with special events, Modification of the adjusted demand to account for addition or deletion of traffic zones (deletion and insertion of centroids). The high level conceptual diagram of the logic structure of the OD tool is described in Figure 14. The diagram shows the correspondence between the main functions. For edition, OD matrices are presented to the user using a spreadsheet. The user can change any value directly typing on the cell or can apply some basic transformation to one cell or more cells as increment/decrement by a factor or adding/subtracting a constant.

The interactive generation of local traversal OD matrices from the global OD matrices is the function required to provide the inputs to the AIMSUN microscopic model of the problem network under analysis. The main input to a route based traffic simulation model is a time dependent origin–destination matrix, each of whose OD_i entries represents the number of trips between the corresponding origin–destination pair for the selected time period i. Usually this information when available concerns the global model of the site being analysed, the WAYFLOW network in the case example of this paper. This is not usually the case when a problem network is selected, unless the problem network has been created in a previous phase and its local OD matrix, or traversal matrix in other words, has been saved in a database containing sets of origin–destination matrices. Therefore the AIMSUN/ISM, in addition to such a database, has the capability to generate interactively such local matrices, combining the versatility of its software architecture with the computational power of the algorithms for traffic assignment and matrix calculations of the EMME/2 software.

The main functions of the OD tool as shown in correspondence with the blocks in Figure 14 are:

1. Automatic translation of the global network model for the site in terms of an EMME/2 model. If link flow counts are available for the time periods corresponding to the global OD matrices for the site, the translated EMME/2 model is prepared to automatically proceed to the adjustment of this OD matrix using the flow counts. An example of this automatic translation is depicted in Figure 15. Figure 15(a) depicts the GETRAM model of a road network and the windows dialogue invoking the translation utility. The GETRAM model, corresponding to the high level representation of the road network required

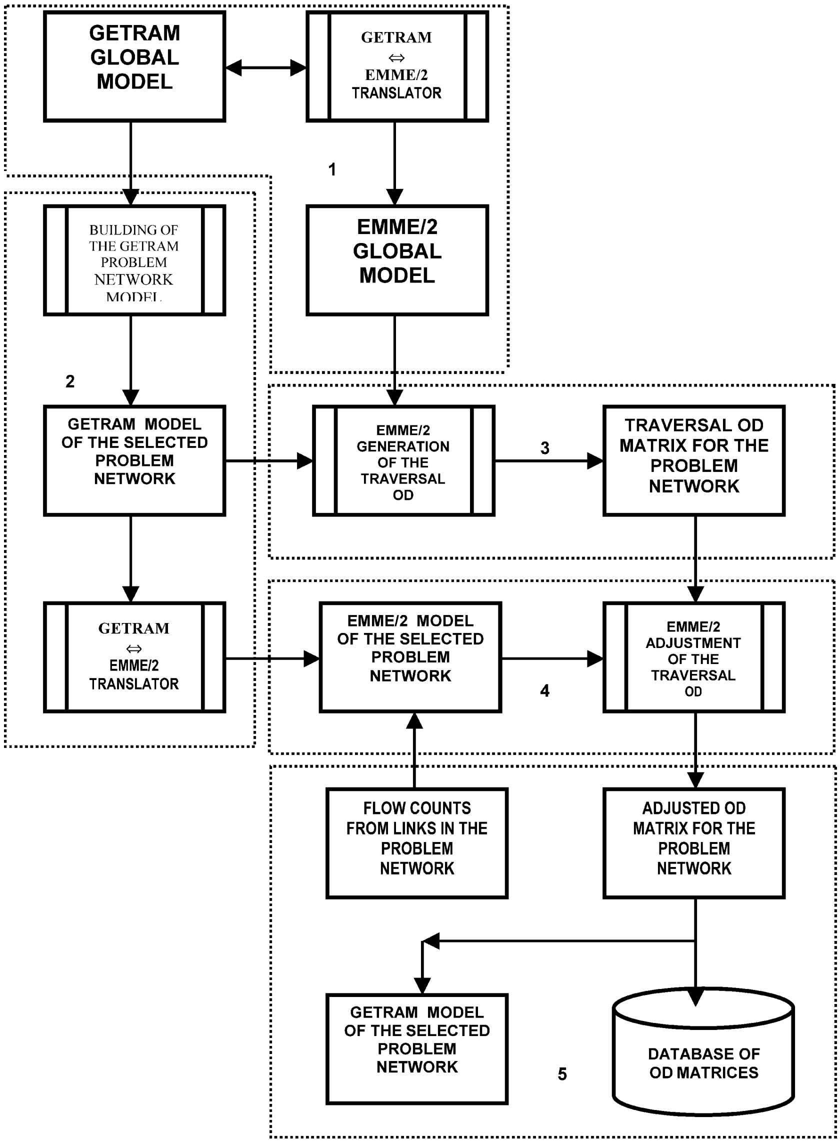
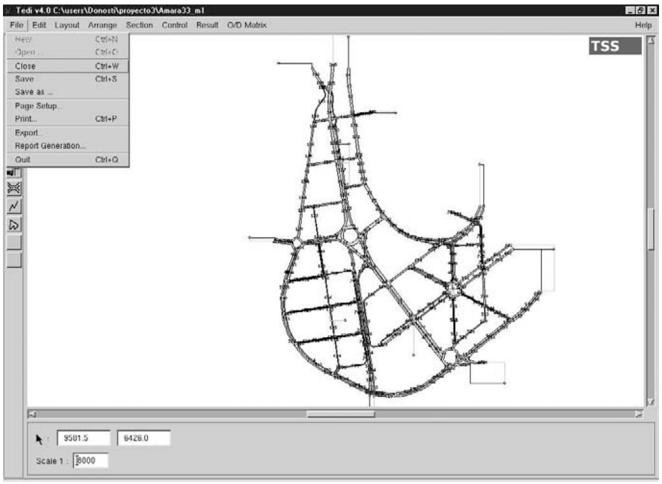


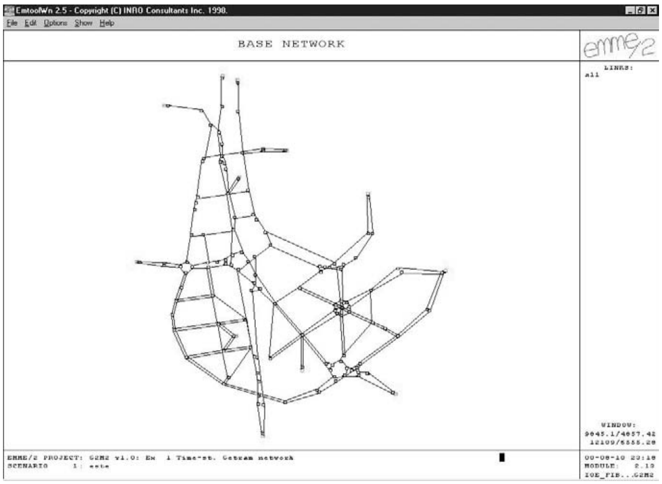
Figure 14. The conceptual structure of the OD tool and its main function.

by the microscopic simulation, is then translated in terms of the aggregated network representation proper of the transport planning models, the EMME/2 in this case, depicted in Figure 15(b). The translation ensures the consistency between levels, micro and macro, of the road network representation.

2. Automatic generation of the subnetwork model for the problem network generated interactively in AIMSUN/ISM, and its translation in terms of an EMME/2 model.



(a)



(b)

Figure 15. Translation from GETRAM to EMME/2.

3. Automatic activation of the suite of programs that calculate the local traversal OD matrix for the problem network.
4. Automatic activation of the adjustment process of the local traversal on basis to the link flow counts for links in the problem network for the time period under consideration.
5. The adjusted local traversal is stored in the database of OD matrices and exported to the AIMSUN model of the problem network as input data for the simulation of the selected scenario in AIMSUN/ISM.

The estimation of time dependent OD matrices for dynamic analysis has, so far, efficient analytical solutions only for specific simple linear networks of motorway type, see for instance the Kalman filtering based approaches in (Chang and Wu, 1994; Nihan and Davis, 1989; Van der Zijp and Hammerslag, 1996). The estimation of dynamic OD matrices for more complex road networks is still an open research topic. From a practical point of view what we propose is a heuristic approximate procedure based on empirical grounds, providing acceptable useful estimates. The heuristic is based on the following assumptions:

1. The network for which the time dependent OD matrix is to be estimated is a subnetwork of a larger network for which an approximate time sliced OD matrix is known.
2. There are available traffic counts on a significant number of links of the selected subnetwork for the time interval of interest.

The procedure consists of three main steps:

1. Starting from a global OD matrix for the whole region for a time horizon T (i.e. the whole day, the peak morning hour, etc.) use additional information on time distribution of trips to generate a set of consecutive OD matrices for smaller time intervals (i.e., for example, for intervals of 30 minutes).
2. Let OD_i be the OD matrix for the ith time interval, assuming that a scenario spanned by a subnetwork of the global network has been selected, the next step extracts the traversal $\mathrm{OD}_i^{\mathrm{T}}$ for the selected scenario for the corresponding time interval, that is the submatrix of the global matrix for the selected subnetwork.
3. Adjust the traversal $\mathrm{OD}_i^{\mathrm{T}}$ from the observed flows for that time interval to estimate the matrix $\bar{\mathrm{O}}\bar{\mathrm{D}}_i^{\mathrm{T}}$ that will become input to the AIMSUN microscopic model for the dynamic simulation.

4.1. TIME SLICING THE GLOBAL OD MATRIX

Figure 16 illustrates graphically the main concepts of this process. The graphics on the left corresponds to the typical view of a global OD matrix as used in traffic assignment. It represents a total number of trips over a time horizon T with an underlying homogeneous behavior. The graphic on the right represents the time variation of the demand. The total number of trips remains constant but they do not behave homogeneously. This representation corresponds to a discretization of the global OD matrix in which the time horizon T has been partitioned into smaller time intervals, and each component of the histogram corresponds to the number of trips for the corresponding interval.

4.2. ESTIMATION OF THE TRAVERSAL OD MATRIX FOR THE SELECTED SCENARIO

To simulate the subnetwork's traffic flows corresponding to the selected scenario for the current period of time, one of the basic data input required is the local

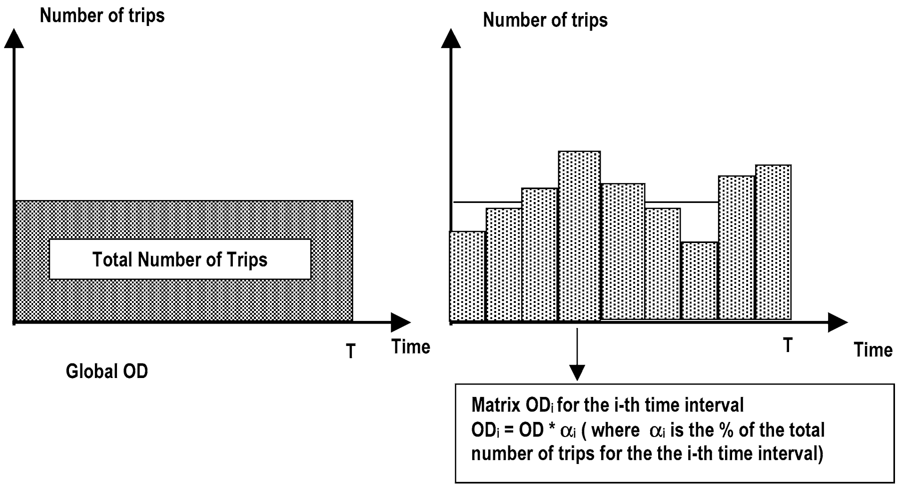


Figure 16. Time slicing a global OD matrix.

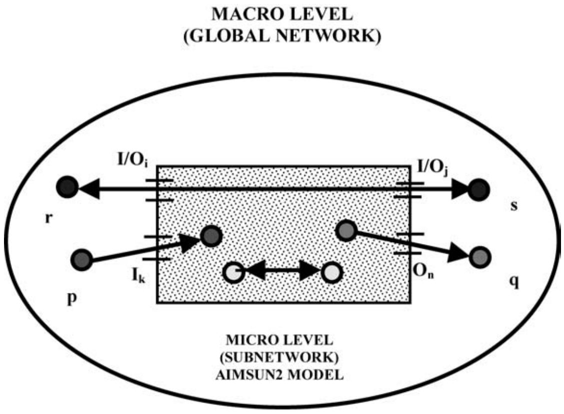


Figure 17. Traversal O/D matrix for a subarea.

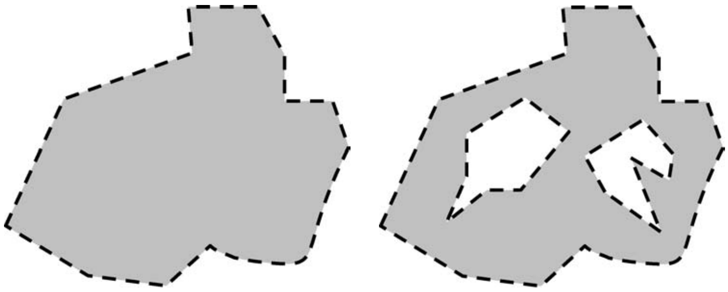
OD matrix for the scenario for that period of time. That is the number of trips t_{ij} between each origin i, and each destination j for each time period. Origins and destinations could lie in the borders of the area spanned by the network, that is the input and output gates defined by the border of the subnetwork, as well as in the area. This is the situation schematized in Figure 17 explained below.

Given an O/D matrix for the whole area and a subnetwork, the proposed procedure starts by calculating the traversal O/D flows between gates that have been defined by the intersection of the border of the subnetwork with the links of the global network. That is, it extracts from the global O/D matrix the local origin–destination submatrix corresponding to the selected subnetwork. This subnetwork corresponds to the scenario graphically selected by the operator, where the traffic conflicts have been identified. This is illustrated graphically in Figure 17.

The so-called traversal matrix is the local O/D matrix for the shaded area inside the rectangle, spanned by a subnetwork of the road network for the whole area. The traversal matrix is composed by the original origins and destinations in the area plus some dummy origins and destinations generated from the input and output gates of the flows into, from and through the area. In Figure 17 I/O_i and I/O_j correspond to the ith and jth input/output gates, which then generate the new dummy nodes, corresponding to the flows form centroid r to centroid s crossing the area. I_k is the kth input gate for the flows with origin at centroid p, outside the area, that finish the trip inside the area, and O_n the nth output gate, for flows generated at a centroid inside the area that leaving the area through this output gate and finish the trip in centroid q outside the area. The generation of traversal matrices is a standard procedure in EMME/2, see (INRO, 1999). Using the additional options auto assignment with its special *traversal operator* does the computation of a traversal matrix. First, the correspondence between gates and zones must be established. The links considered as *in-gates* are all the outgoing connectors from the centroids located in the selected scenario, as well as all the links that enter the scenario boundaries. The links considered as *out-gates* are all the incoming connectors to the centroids located in the scenario, as well as all the links that exit the scenario boundaries. User data items (i.e. **ul3**, user defined attribute in EMME/2) can be used to hold the gate information. For that purpose, it must be initialized to 0, and then prepared as follows:

- All centroid connectors within the scenario are defined as directional gates: all outgoing connectors, which have the centroid as I-node, are tagged with the positive centroid number (in-gates) and all the incoming connectors, which have the centroid as J-node, are tagged with the negative centroid number (out-gates). This can be done systematically by using the network calculator (Module 2.41 of EMME/2).
- All the streets that cross the scenario boundary are assigned centroid numbers and are defined as directional gates, with the convention that if a street is two-way, the same centroid number is assigned to both corresponding links. All links, which enter the scenario boundary, are tagged with a positive centroid number (in-gates) and all links, which exit the scenario boundary, are tagged with a negative centroid number (out-gates). This can be done by using the graphic worksheet (Module 2.12 of EMME/2), where the links crossing the scenario boundary can be identified easily.

The utilities implemented in the AIMSUN/ISM ODTool perform all these functions automatically for the problem network under study once it has been graphically defined as described in the introductory section. The most important part are the functions relative to the automatic identification and definition of the *gating system* for the GETRAM subnetwork model of the problem network for its translation in the required terms of the EMME/2 traversal calculation mode. The function assumes that the basic input defining graphically a subnetwork is a set of



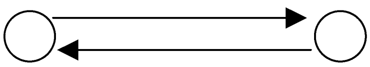
(a) (b)

Figure 18. A connected problem network defined by a single polygonal line (a) and by several polygonal lines (b).

closed polygonal lines that define a connected area, as shown in the two examples in Figure 18.

Just identifying which nodes fall inside the problem network can then derive the elements making up the subnetwork. The rules of the generation of the EMME/2 model of the problem network for the automatic calculation of the traversal OD matrix, illustrated in Figure 19, are the following:

1. The sections of the subnetwork must be the sections of the global area that have their starting node and/or their ending node with coordinates lying in the interior of the region that defines the problem network. All the attributes of the sections in the subnetwork are inherited from the corresponding ones in the global area.
2. The nodes of the subnetwork model are those with coordinates lying in the problem network.
3. The sections of the subnetwork that have starting node with coordinates in the problem network define an exit from the subnetwork. The sections of the subnetwork that have ending node with coordinates in the problem network define an entry to the subnetwork. Inputs and outputs can gather forming a *gate* only if their sections were opposite in the global network model (i.e. having common starting/ending nodes:



"Opposite sections or links"

Centroids of the global network model with at least a connector attached to a node with coordinates lying in the problem network correspond also to a single gate of the subnetwork model. If all the connectors of the centroid have attachment points lying in the problem network then, the centroid must be considered as a *centroid fully interior* to the problem network. In this case the gate number matches the centroid number assigned to the centroid in the global area (see, in Figure 19, centroid 23). Otherwise, i.e. if there exist connectors with attachment not in the

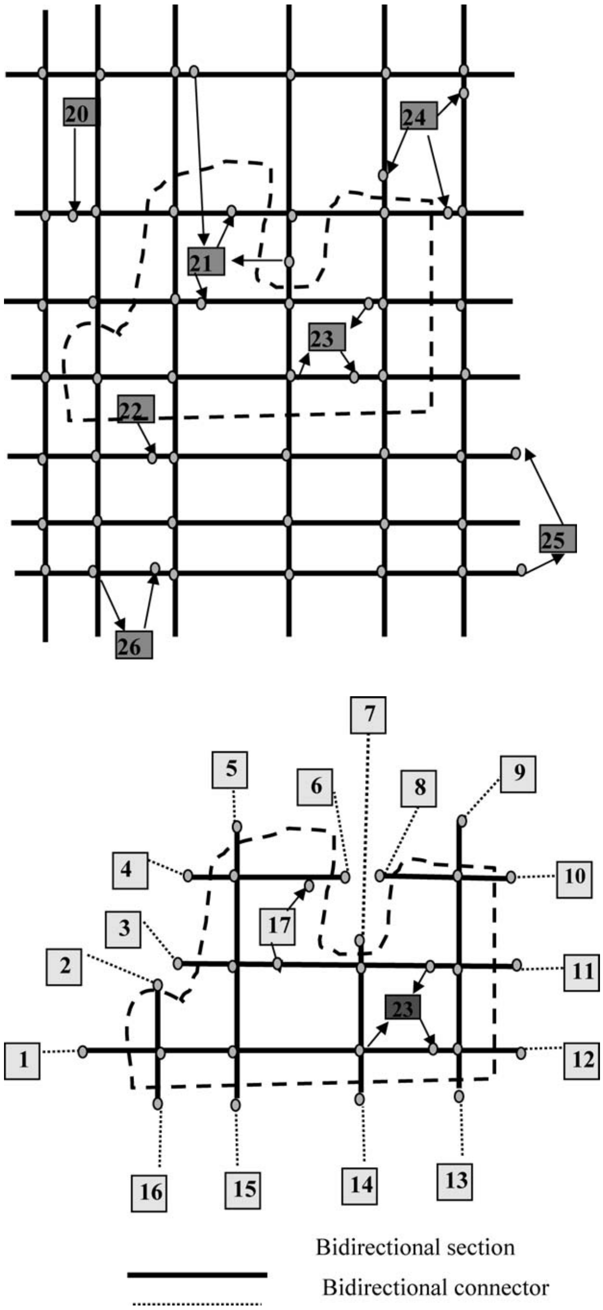

20
21
24
23
22
25
26
7
5
6
8
9
4
10
17
3
11
2
23
1
12
16
15
14
13
Bidirectional section
Bidirectional connector

Figure 19.

problem network, then the gate number can be other than the centroid number in the global area (see, in the figure centroid number 21, which corresponds to gate number 17 in the subnetwork model).

4. Gate numbers can be given arbitrarily to the gates of the subnetwork, excluding the case of gates corresponding to *centroids fully interior* to the problem network, which must have gate number equal to the centroid number in the global area.
5. There cannot be duplicates in numbering the gates of a subnetwork model.
6. Gates are attached to the corresponding starting or ending nodes of physical sections by means of connectors. The connectors emerging or incoming to gates of the subnetwork model.

4.3. ADJUSTMENT OF THE TRAVERSAL MATRIX

The traversal matrix has been extracted from a global OD matrix whose information corresponds to an average long-term representation of trip patterns; therefore it could have significant deviations with respect to the actual trip patterns for the time interval under consideration. If information is available about the current traffic flows on the links of the subnetwork, or at least on a significant number of links, then this information can be used to adjust the local OD matrix and get a better representation of the trip patterns.

The core of this heuristic is an adjustment method based on a bilevel optimization method. The algorithm can be viewed as calculating a sequence of O/D matrices that consecutively reduce the least squares error between traffic counts coming form detectors and traffic flows obtained by a traffic assignment. The calculation of the traversal matrix for a subarea requires information about the routes used by the trips contained in the O/D matrix (d_{ij}). It requires the definition of the route and the trip proportions relative to the total trips d_{ij} used on each route originating at zone I and ending at zone j. This information is really difficult to handle and store in traffic databases, taking into account that the number of routes connecting all origin–destination pairs on a connected network can grow exponentially with the size of the network. This is the reason to use a mathematical programming approach based on a traffic assignment algorithm, solved at each iteration without requiring the explicit route definition that computes the traversal matrix during the network-loading phase of the algorithm.

The implementation of the heuristic, which follows the methods proposed by Florian and Chen (1995), or Spiess (1990), is based on the available interfaces between GETRAM/AIMSUN and EMME/2, and the utilities implemented in EMME/2 macros. Once the traversal is computed the adjustment can be done in a standard way using the EMME/2 macro **demadj.mac**, which is activated automatically by AIMSUN/ISM. The adjusted matrix is then imported by AIMSUN/ISM into AIMSUN through the corresponding dialogue.

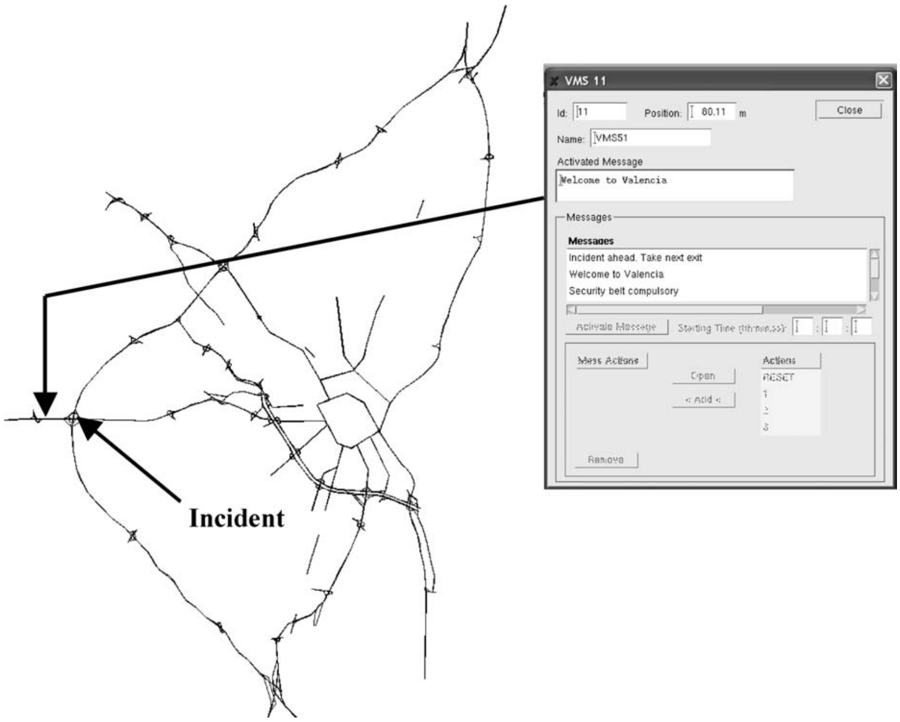


Figure 20.

4.4. AN EXAMPLE OF SCENARIO ANALYSIS: THE EVALUATION OF THE IMPACT OF A TRAFFIC MANAGEMENT STRATEGY

Figures 20 and 21 depict an example of the use of the Scenario Analysis Module in AIMSUN/ISM for testing and evaluating a traffic management strategy.

Figure 20 presents the scenario, that is the simulation model of a road network (the bypass of the city of Valencia in Spain) identifying the location of a variable message panel. The strategy defined by dialogues similar to the ones described in Figures 12 and 13, defines for the VMS panel a set of predefined messages, each of them associate to a set of potential actions. The strategy to evaluate in this case corresponds to the re-routing actions in the case an incident of a certain severity (measured in terms of the number and length of blocked lanes).

Figure 21 depicts the results of the evaluation process. The detector upstream of the incident location provides the information required by an Automatic Incident Detection System (Barceló et al., 2002a) to detect the incident. After the detection of the incident the strategy activates the planned action displaying at the VMS panel the information that there is an incident ahead, and recommending to take the next exit. To help the analyst the simulator has the option of highlighting in different colors the vehicles depending on whether they will leave the motorway by the next exit or not. This feature is also shown in Figure 21. Those vehicles,

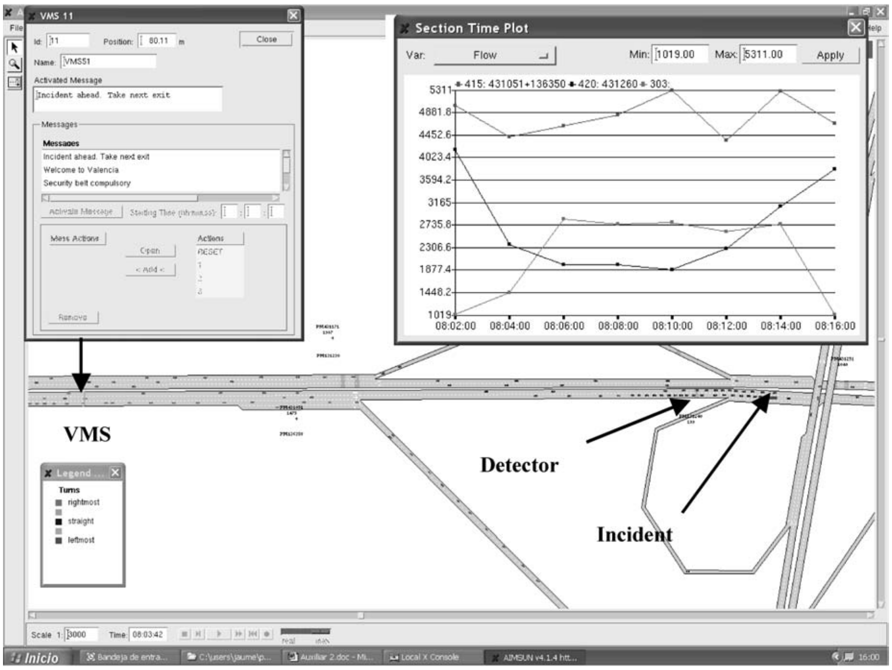


Figure 21.

which can still reach their destinations if they follow the recommendation, will be the candidates to leave the motorway by the next exit. From the modeling point of view this raises a behavioral question, how will the drivers react to the given information. In absence of behavioral models representing appropriately these reactions, the analyst can define alternative scenarios assuming different levels of accomplishment. The graphics in the right upper corner of Figure 21 displays the simulated response for one of such scenarios. The upper red curve represents the evolution of the traffic flow in the main section, where the VMS is located. The lower green curve represents the flow in the exit ramp, showing how it increases after the message is displayed in the VMS, until reaching a stationary value lasting for the duration of the incident, and decreasing after the incident is cleared out. The medium blue curve represents the evolution of the traffic flow in the section after the exit ramp where the incident has occurred. As should be expected the flow decreases until reaching a stationary value, and starts to increase after the incident is removed. The flow values provide the figures for a quantitative analysis on which to assess the proposed management strategy.

5. Concluding Remarks

This paper has presented two contributions to microscopic traffic simulation and the use of microscopic traffic simulation to support traffic management decisions.

The first contribution has presented an extension of the traditional microscopic simulation paradigm in which, instead of a pure stochastic emulation of traffic flows on a road network, vehicles are assigned to paths from origins to destinations according to stochastic route choice models. This extension is a fundamental requirement to properly evaluate by simulation most of the more relevant intelligent transport systems, as, for example, all those based on management strategies trying to redistribute the flows in the network by re-routing the vehicles according to the prevailing traffic conditions.

The second contribution proposes a methodology for the analysis of traffic scenarios for conflicting situations using the microscopic simulation, and describes how this methodology has been implemented in a new software platform AIMSUN/ISM, that combines the improved AIMSUN simulator with two modules. A module to generate the traffic scenarios to be simulated, the Scenario Analysis Module, and another one to interactively generate the required mobility patterns, in terms of origin–destination matrices, required for the simulation of the scenarios. AIMSUN/ISM has been successfully implemented and tested as part of the ISM (Intermodal Strategy Manager) project in Hessen, Germany, in the framework of the WAYFLOW Program.

References

Banks, J. (ed.): 1998, *Handbook of Simulation: Principles, Methodology, Advances, Applications and Practice*, Wiley, New York.

Barceló, J.: 2000, The role of simulation in advanced traffic management systems, in: *Proc. of the 7th World Conf. on ITS*, Torino.

Barceló, J.: 2001, GETRAM/AIMSUN: A software environment for microscopic traffic analysis, in: *Proc. of the Workshop on Next Generation Models for Traffic Analysis, Monitoring and Management*, Tucson, AZ, September.

Barceló, J. and Casas, J.: 1999, The use of neural networks for short-term prediction of traffic demand, in: A. Zeder (ed.), *Transportation and Traffic Theory*, Pergamon.

Barceló, J. and Casas, J.: 2003, Dynamic network simulation with AIMSUN, Preprints of the *Proc. of the Internat. Symposium on Transport Simulation*, Yokohama, Kluwer, in press.

Barceló, J., Casas, J., and Ferrer, J. L.: 1999a, Analysis of dynamic guidance systems by microsimulation with AIMSUN2, in: *Proc. of the 6th World Conf. on ITS*, Toronto.

Barceló, J., Casas, J., Ferrer, J. L., and García, D.: 1999b, Modeling advanced transport telematic applications with microscopic simulators: The case of AIMSUN2, in: W. Brilon, F. Huber, M. Schreckenberg and H. Wallentowitz (eds), *Traffic and Mobility, Simulation, Economics, Environment*, Springer, Berlin.

Barceló, J., Ferrer, J. L., and Martin, R.: 1999c, Simulation assisted design and assessment of vehicle guidance systems, *Internat. Trans. Oper. Res.* **6**(111), 123–143.

Barceló, J., Ferrer, J. L., Grau, R., Florian, M., Chabini, I., and Le Saux, E.: 1995, A route based variant of the AIMSUN2 microsimulation model, in: *Proc. of the 2nd World Congress on Intelligent Transport Systems*, Yokohama.

Barceló, J., Ferrer, J., Casas, J., Montero, L., and Perarnau, J.: 2000a, Microscopic simulation with AIMSUN for the assessment of incident management strategies, in: *Proc. of the e-Safety Conference*, Lyon, September.

Barceló, J., García, D., and Kirschfink, H.: 2000b, Scenario analysis a simulation based tool for regional strategic traffic management, in: *Traffic Technology International 2001*, Special Issue.

Ben-Akiva, M. and Bierlaire, M.: 1999, Discrete choice methods and their applications to short term travel decisions, in: *Transportation Science Handbook*, Kluwer.

Cascetta, E., Nuzzolo, A., Russo, F., and Vitetta, A.: 1996, A modified logit route choice model overcoming path overlapping problems, in: *Proc. of the 13th Internat. Symposium on Transportation and Traffic Flow Theory*, Pergamon.

Chang, G. and Wu, J.: 1994, Recursive estimation of time-varying origin–destination flows from traffic counts in feeway corridors, *Transport. Res. B* **28**(2), 141–160.

Florian, M. and Chen, Y.: 1995, A coordinate descent method for the bi-level OD matrix adjustment problem, *Internat. Trans. Oper. Res.* **2**(2), 165–175.

Florian, M., Mahut, M., and Tremblay, N.: 2001, A hybrid optimization–mesoscopic simulation dynamic traffic assignment model, in: *Proc. of the 2001 IEEE Intelligent Transport Systems Conference*, Oakland, pp. 120–123.

Gabbard, J. F.: 1991, Car-following models, in: M. Papageorgiou (ed.), *Concise Encyclopedia of Traffic and Transportation Systems*, Pergamon, Oxford.

Gipps, P. G.: 1981, A behavioral car-following model for computer simulation, *Transport. Res. B* **15**, 105–111.

Gipps, P. G.: 1986, A model for the structure of lane-changing decisions, *Transport. Res. B* **20**, 403–414.

INRO Consultants: 1999, EMME/2 User's Manual, Software release 9.0.

Mahut, M.: 2000, Behavioral car following models, PhD Thesis, Chapter 3, Centre de Recherche sur les Transports, CRT, Université de Montréal.

Nihan, N. L. and Davis, G. A.: 1989, Application of prediction–error minimization and maximum likelihood to estimate intersection OD matrices from traffic counts, *Transport. Sci.* **23**, 77–90.

Spiess, H.: 1990, A gradient approach for the OD matrix adjustment problem, Publication No. 693, Centre de Recherche sur les Transports, Université de Montréal.

TSS: 2002, *User's Manuals*, Transport Simulation Systems, GETRAM/AIMSUN Version 4.1.

Van der Zijp, N. J. and Hammerslag, R.: 1996, Improved Kalman filtering approach for estimating origin–destination matrices for freeway corridors, *Transport. Res. Record* **1443**, 54–64.