Chapter 5
Traffic Simulation with Aimsun

Jordi Casas, Jaime L. Ferrer, David Garcia, Josep Perarnau, and Alex Torday

5.1 Introduction

5.1.1 Background and Overview

Originally the focus of a long-term research programme at the University of Catalonia (UPC), the Aimsun transport modelling software is now in its sixth major commercial version. Having outgrown the stated aim of the original AIMSUN acronym 'advanced interactive microscopic simulator for urban and non-urban networks' (Ferrer and Barceló, 1993; Barceló et al., 1994, 1998a), the software now includes macroscopic, mesoscopic and microscopic models and is simply known as 'Aimsun' (Aimsun, 2008).

Expanding in response to practitioners' requirements, Aimsun 6 has come to encompass a collection of dynamic modelling tools. Specifically, these include mesoscopic and microscopic simulators and dynamic traffic assignment models based on either user equilibrium or stochastic route choice. From a practitioner's standpoint, macroscopic modelling plays an increasingly important role in the area of demand data preparation. However, in line with the scope of this book, this chapter focuses on Aimsun's dynamic modelling capabilities.

The primary areas of application for Aimsun are offline traffic engineering and, more recently, online (real-time) traffic management decision support. In either case, the use of Aimsun or Aimsun Online aims to provide solutions to short and medium-term planning and operational problems for which the dynamic and disaggregate models described in this chapter are extremely well suited. Strategic planning is an

J. Casas (✉)
TSS – Transport Simulation Systems, S.L, Passeig de Gràcia 12, 08007 Barcelona, Spain and Universitat de Vic, Carrer Sagrada Familia 7, 08500, Vic, Spain
e-mail: casas@aimsun.com

J.L. Ferrer, D. Garcia, J. Perarnau, and A. Torday
TSS – Transport Simulation Systems, S.L, Passeig de Gràcia 12, 08007 Barcelona, Spain
e-mails: jlferrer@aimsun.com; david@aimsun.com; josep@aimsun.com; torday@aimsun.com

J. Barceló (ed.), *Fundamentals of Traffic Simulation*,
International Series in Operations Research & Management Science 145,
DOI 10.1007/978-1-4419-6142-6_5, © Springer Science+Business Media, LLC 2010

adjacent realm for which more aggregate and/or static models continue to be very suitable. There are important interfaces between those two realms at the level of methodology (effect on demand of lasting changes to the effective capacity) and technology (importing from and exporting data to strategic planning software) and we will comment on those issues further in the coming sections.

5.1.2 Development Principles

The remainder of this chapter will provide details on the models inside Aimsun as they currently stand. Inevitably, this description can only be a snapshot of what is currently available with very limited references to ongoing developments. The fast-paced evolution of this software category in general, and of Aimsun in particular, almost guarantees that some aspects of our description will be obsolete or, at best, incomplete soon after publication. It is therefore worth providing the reader with a brief outline of the overarching principles that will continue to inform the development direction of our transport modelling software platform beyond its sixth major release.

- *Integration*: The steadily improving volume and quality of traffic data, availability of computing resources and, perhaps most importantly, practitioner's expertise have given rise to sophisticated and rich methodological frameworks for dynamic modelling. A key exponent of this tendency is the integrated corridor management initiative described in Alexiadis (2007) and put into practice throughout North America and elsewhere (Stogios et al., 2008; Torday et al., 2009). Models are steadily growing in size, complexity and detail; and feedback loops between demand and supply changes are explicitly acknowledged in modelling studies. This development naturally calls for elimination of duplicate information that, in that context, represents wasted effort and risk of error.

  Concretely, in the case of Aimsun, the information shared by all models is network topology, OD demand and time-dependent shortest (or cheapest) paths and their respective travel times (or costs). Demand and paths/path costs are not merely a shared input; it is rather the case that the application of one model can produce those outputs in a format that is directly exploitable by another Aimsun model. The advantages of this approach are further elaborated in the case studies discussed in Section 5.7. The underlying principle is to integrate everything that can and should be shared between all the models to enable sophisticated workflows that involve sequential, iterative or even concurrent application of two or more models. The other implication of the principle is *completeness*; put simply, this translates into 'integrate everything that is required to meet a modelling study's objectives comprehensively'.
- *Modularity*: This refers to the breaking down of processes or tasks into elementary units, allowing their consistent and easy re-use within larger processes. Examples of how this principle has been applied in Aimsun include sharing

the gap acceptance or traffic management modules between the mesoscopic and microscopic simulators. Perhaps more crucially, an insistence on modularity allows Aimsun users to decouple the link dynamics models (i.e. the mesoscopic and microscopic simulators) from the process by which traffic is assigned dynamically to various routes. This gives rise to two new combinations of models with potentially very interesting and useful applications. Firstly, the combination of mesoscopic simulation with dynamic traffic assignment by means of locally applied stochastic route choice can be a great tool for faster than-real-time simulation of non-recurring incidents – especially so when it is also possible to specify that drivers away from the incident's influence area will continue to follow established routes (the results of a previous dynamic equilibrium assignment). Secondly, the combination of dynamic user equilibrium and microscopic simulation can shed a lot of light onto town-centre modelling in which modal shift policies (pedestrianisation, dedicated bus lanes, public transport pre-emption schemes) bring about lasting changes to network utilisation and in which pedestrians, public transport and private vehicles interact at stops and crossings creating mutual dependencies that affect capacity in complex ways. The underlying principle is to break models down into basic ingredients such that practitioners can re-combine these ingredients as appropriate into new methodological 'recipes' (best practice).

- *Scalability*: Computing hardware continues to improve every year. ITS and other developments related to communication follow at an equally fast pace, leading to a parallel improvement in the availability and quality of traffic data. The two factors combined push the envelope: dynamic modelling is now considered desirable and useful in areas such as the entire area of lower Manhattan, New York; Montreal in Quebec or all of Singapore. Fortunately, it is also feasible using mesoscopic simulation or multi-threaded implementations (Barceló et al., 1998b) of micro-simulation or a combination of both (Barceló et al., 2006). The challenge in the years to come will be to keep apace with changing intensive computation paradigms whilst responding to ever more stringent performance requirements imposed by users. Efficient software design is, in our opinion, a prerogative in that regard.
- *Interoperability and extensibility*: The proliferation of ITS systems on the one hand and GIS/mapping and 3D technologies on the other means that creating an accurate model of a road network is becoming much more difficult and quite a lot easier at the same time. Variable speed control, dynamic lane assignment, congestion pricing and adaptive signal control are examples of novelties that need to be captured accurately in models. Conversely, high-quality maps and GIS data simplify model building compared to a few years ago; as for 3D building information, it is making the prospect of large-scale virtual reality viable. Whatever the case may be, all these developments call for *interoperability*, that is the ability to exchange data with other applications in a variety of formats, and *extensibility*, that is the ability for users to programme custom extensions relatively easily. Again, we note that the ability to respond to these challenges rests, to a large

degree, with efficient software design although *standards* play an important role here as well.

The remainder of this chapter is organised as follows:

- Model-building principles in Aimsun
- Fundamental core models: car following and lane changing
- Dynamic traffic assignment
- Calibration and validation of Aimsun models
- Extended modelling capabilities: working with external applications
- Selected overview of advanced case studies and applications
- Modelling details of advanced case studies

5.2 Model-Building Principles in Aimsun

Building a transport simulation model with Aimsun is an iterative process that comprises three steps:

- Model building, that is, the process of gathering and processing the inputs to create the model;
- Model verification, calibration and validation, that is the process of confirming that implementation of the model logic is correct; setting appropriate values for the parameters and comparing the outputs of the model to corresponding real-world measurements in order to test its validity;
- Output analysis, that is the exploitation of model outputs in line with the overall objectives of the modelling study.

The following three sub-sections provide further detail on these steps.

5.2.1 Model Building

Building an Aimsun model requires two types of information:

- *supply data*, that is everything related to the infrastructure and services that allow goods and people to travel coded as a graph of sections and turns with associated attributes;
- *demand data*, that is the mobility needs, coded as a set of OD matrices, one for each vehicle type and time interval.

Calibration and validation bring about additional data requirements at both the supply and the demand level. This can include information about the types of vehicles (e.g. acceleration characteristics), drivers (e.g. level of compliance with speed limit) and actual levels of traffic through the network for known levels of demand. Accordingly, Aimsun supports the concept of real data sets and allows the user to manage them.

5.2.1.1 Supply Data

Supply data includes all the information related to the transportation network and services, such as

- geometric and functional specification of the road network;
- traffic control;
- public transport services;
- other (for instance, fleet vehicles).

5.2.1.2 Geometric and Functional Specification of the Road Network

Geometric information that is needed to build an Aimsun model are the following:

- road shape;
- number of lanes;
- reserved lanes;
- turnings allowed at the end of each section – from which lane(s) to which lane(s), together with stop points and priorities between conflicting movements;
- pedestrian crossings.

Functional attributes (some of which present in all geographic information systems) depend on the level (micro, meso or macro) of simulation (Table 5.1).

Table 5.1 Physical parameters required for each model type in Aimsun

	Macro		Meso		Micro	
Parameter	Required	Optional	Required	Optional	Required	Optional
Maximum speed	X		X		X	
Capacity	X			X[a]		X[a]
User-defined costs		X		X		X
Volume delay function	X					
Visibility distance at yield intersections					X	
Slope	X		X		X	

[a]Can be used as an additional parameter for route choice

The process of coding the network uses as its basis a file in one of the following formats imported into Aimsun:

- aerial images, such as PNG, JPG, BMP, GIF, SVG, SID, ECW, JP2 and TIFF;
- 3D models, such as 3dsmax 3DS and Wavefront OBJ;
- CAD, such as AutoCAD DWG or DXF, and Microstation DGN;
- GIS, such as ESRI SHP, MapInfo TAB or MIF, OpenGIS GML, GPX and Google KML;
- digital maps, such as Navteq maps files;
- input files for other transport modelling or signal optimization software applications, such as Emme, SATURN, CONTRAM, VISUM, VISSIM, PARAMICS, TRANSYT and SYNCHRO.

The amount of information that can be automatically converted, and thus the amount of manual refinement that is needed, depends on the type of model that must be built (macro, meso or micro) and on the format of the input file. The file formats above are listed roughly in the order of ascending utility for modelling purposes (i.e. the most useful format is listed last).

For example aerial images, CAD files and 3D models do not carry any kind of topology or functional information, so they can only be used as background to guide manual network building; some image formats do not even provide geographical location or scale.

GIS files, maps and other traffic simulation software files are the best formats because they include both geometry and additional attributes associated with each entity, so the import process automatically creates a complete Aimsun network; but even in this case a subsequent manual refinement is needed, for example to locally adjust node details and to input a parameter required for Aimsun simulations for which there is no equivalent in the third-party software.

5.2.1.3 Traffic Control

For microscopic and mesoscopic simulation, traffic control plans during the simulation period are needed for all signalized intersections and for any ramp meterings included in the model.

Traffic signals can be fixed, i.e. defined in advanced and remain unchanged during the simulation period, or actuated, whereby the control plan is dynamically modified depending on measured traffic conditions.

For each fixed control plan, Aimsun requires the following information:

- start time and duration of the control plan;
- cycle length;
- amber/yellow duration;
- turnings associated with each signal group (including pedestrian movements);

- timings of each signal group;
- offset relative to other control plans.

Actuated control plans can be modelled using virtual detectors which are the simulation equivalents of real-world loop detectors or other similar devices. As for the control logic, this can be specified in Aimsun, if the real controller is compliant to the NEMA standard. With SCATS, UTOPIA, VS-PLUS, SICE and SCOOT, a data interchange interface can be used; otherwise the control logic can be emulated by programming a custom API (application programming interface) extension which exchanges data between Aimsun and the software implementation of the corresponding controller logic.

5.2.1.4 Public Transport

In order to include public transport into the model, the following information must be provided:

- route of each line;
- stop locations;
- departure frequency or timetable;
- stop-time mean and deviation. This can be global or, as an option, a function of the line, stop and time of the day.

5.2.1.5 Demand Data

Traffic demand is input into Aimsun in the form of either (time-dependent) OD matrices or traffic states (micro only).

If OD matrices are used (recommended for micro and required for meso), it is necessary to have the zoning of the modelled area to correctly place the centroids and their connections. The placement of centroid connections has to be carefully studied so that the entry and exit rate into and from the model is as realistic as possible. To minimize this distortion, internal centroids should never be connected to main streets but to local roads and preferably at nodes rather than to specific sections.

In order to reproduce traffic patterns and fluctuations faithfully, it is advisable to input separate OD matrices for different vehicle types using small time slices (possibly 15 min).

Similarly with all models of this category, recent and reliable matrices are a prerequisite: to that end, a matrix adjustment using traffic counts is often helpful for improving the quality of an old matrix if a more recent one is not available – provided that no big changes have occurred in land use.

OD matrices can be pasted into Aimsun via a simple copy operation in Microsoft Excel or be read directly from an ASCII file or any database via an ODBC connection.

If traffic states are used (possible only for micro-simulation), users need to specify the input flows for all entrance sections and the turning percentages at each node where more than one turning is possible.

Simulating with traffic states is generally not recommended. Firstly, for some network configurations, it is possible that the sampling process at nodes will lead to one or more particular vehicles becoming ‘trapped in’ the network (that is they move around continuously and never exit); so it is acceptable only for small networks whose connectivity does not lend itself to loops. Secondly, for all types of networks, traffic states cannot be used to simulate a future scenario in which a change to the supply or the demand may create different route utilization.

5.2.2 Model Verification, Calibration and Validation

Before starting to modify model parameters in order to calibrate the model, the user must be sure that there are no specification errors that affect the model logic and therefore simulation results.

Verification consists in assuring that the model has been correctly edited in Aimsun, checking network geometry, control plans, management strategies and traffic demand, and verifying that the model description corresponds to the objectives of the study.

Aimsun provides a tool that can automatically detect errors in supply definition, such as a section where not all the lanes at the beginning or at the end are connected or an OD pair with trips but no feasible path.

Verification of traffic demand is done through a manual comparison with traffic counts wherever possible; for example the total trips generated and attracted by a zone must be compared with the counts of the sections to which the corresponding centroid is connected.

An important check is to verify that the model is suitable for the objectives of the study; the model must include all the area that might be influenced by future changes being modelled; the boundaries must be free of congestion; if rerouting strategies are simulated, then alternative paths must be possible in the network being modelled; OD matrices should be time sliced so as to reproduce traffic demand dynamics correctly and the study time frame must extend beyond (earlier than) the peak hour to avoid starting the simulation in an oversaturated condition.

Calibration is an iterative process that consists of changing model parameters and comparing model outputs with a set of real data until a predefined level of agreement between the two data sets is achieved.

Which output needs to be compared depends on the type of model (macro, meso or micro), the objective of the study and the type of network. The most significant measures for a highway model are the relationship between speed/flow/density, lane utilization and congestion propagation. For an urban model, queue length, queue discharge speed and levels of services in large and/or more complex networks traffic flows and travel times become important as well.

It is important to emphasize that traffic counts are generally not sufficient for calibrations; as is well known (reference to fundamental equation), the same flow value can be reached in congested and uncongested conditions, so at least one more measure (speed, occupancy, etc.) is needed.

Once calibrated, the model must be validated comparing its outputs with a set of real data different from that used for calibration; if a predefined level of agreement between simulation and real-world data is achieved, the model can be considered valid, and thus suitable for studying future scenarios, subject to there being no changes to the real-world network that invalidate the model assumptions. The same considerations on output comparison made for calibration apply to validation as well.

For comparison purposes, Aimsun provides an interface capable of reading real data stored in ASCII files (space-, comma-, tab or semicolon-separated values), ODBC databases and GPX (GPS exchange format) files, linking them to model objects by id, external id or name.

5.2.3 Output Analysis

The dynamic (meso and micro) models produce time series in which each value is the aggregation of data collected during a regular interval defined by the user.

Mesoscopic simulations produce, for each section

- flow (also available for turnings)
- density
- speed
- travel time and delay time
- queue length

Microscopic simulations produce

- flow;
- density;
- speed and harmonic speed;
- travel time and delay time;
- queue length;
- stops and stop time;
- pollution and fuel consumption;
- trajectory data.

It is also possible to collect data at the level of 'streams' (a set of sections selected by the user) as well as being able to output global, network-wide values of the above outputs.

5.3 Fundamental Core Models: Car Following and Lane Changing

The core models in Aimsun deal with individual vehicles, each vehicle/driver having behavioural attributes assigned to them when they enter the system; those attributes remain constant during the whole trip. The difference between the core models at the mesoscopic and microscopic levels relates to the level of abstraction and to the process employed to update each vehicle's status. Accordingly, in what follows, we describe separately two sets of fundamental core models: microscopic behavioural models and mesoscopic behavioural models.

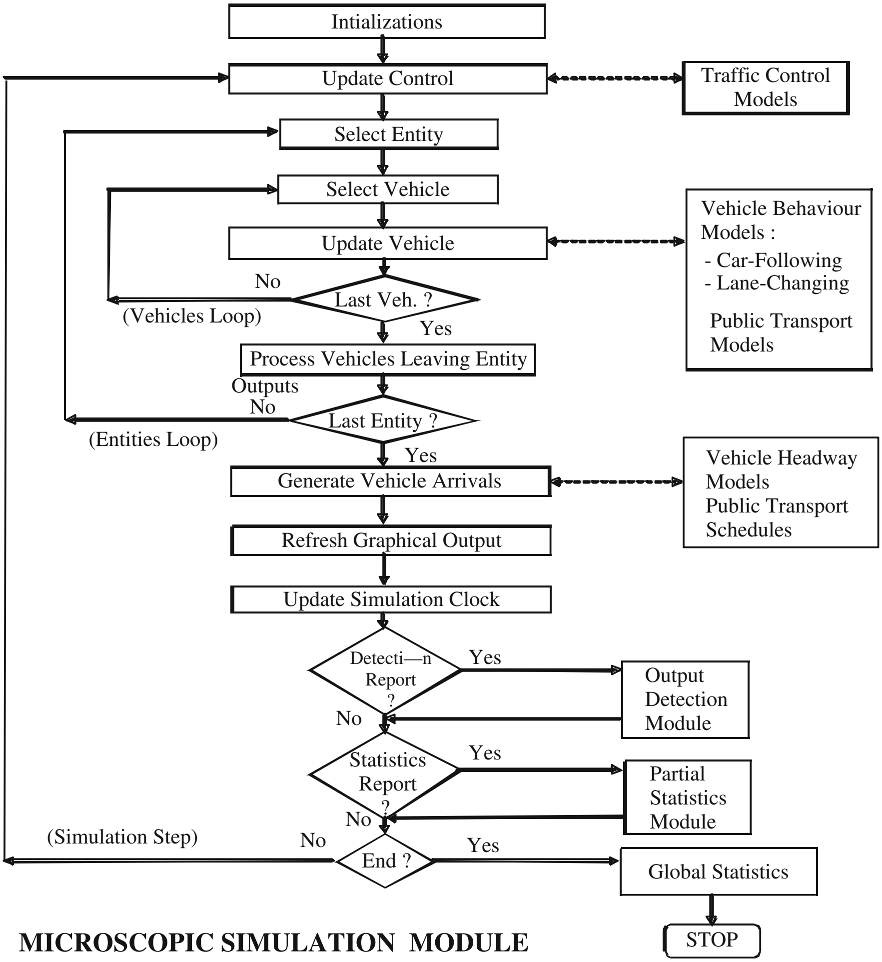


Fig. 5.1 The microscopic simulation process in Aimsun

5.3.1 Microscopic Logic of Simulation Process

The logic of the microscopic simulation process in Aimsun is illustrated in Fig. 5.1. It can be considered as a time slice-based simulation with an additional scheduled event calendar. At each time interval (simulation step), the simulation cycle updates the unconditional events scheduling list (i.e. events such as traffic light changes which do not depend on the termination of other activities). The 'Update Control' box in the flow chart represents this step. After this updating process, a set of nested loops starts to update the status of the entities (road sections and intersections) and vehicles in the model. Once the last entity has been updated, the simulator performs the remaining operations such as inputting new vehicles and collecting new data.

5.3.2 Mesoscopic Logic of simulation Process

The mesoscopic model in Aimsun works with individual vehicles but adopts a discrete-event simulation (Law and Kelton, 1991) approach, in which the simulation clock moves between events and there is no fixed time slice. An *event* is defined as an instantaneous occurrence that may change the state of the traffic network, i.e. the number of vehicles in sections and lanes, the status of the traffic signals etc. Events can be *scheduled* (known in advance to occur at a particular time in the simulation) or *conditional* (added to the event list dynamically during the simulation whenever some logical condition is satisfied). Specifically, a mesoscopic simulation includes the following types of events:

- Vehicle generation (vehicle entrance);
- Vehicle system entrance (virtual queue)
- Vehicle node movement (vehicle dynamics)
- Change in traffic light status (control)
- Statistics collection (outputs)
- Matrix change (traffic demand)

These events model the vehicle movements through sections and lanes by using a simplification of the car-following, lane-changing and gap-acceptance models used in the microscopic simulator. Nodes, on the other hand, are modelled as queue serves. All events have an associated time and a priority. Both these attributes are used to sort the event list. For example, events related to a change in the status of a traffic light or a new vehicle arrival are going to be treated before events that relate to statistics collection or vehicle movements inside a node.

5.3.3 Modelling Microscopic Vehicle Movement

In the Aimsun micro-simulator, during a vehicle's journey along the network, its position is updated according to two driver behaviour models termed 'car following' and 'lane changing'. The premise behind the models is that drivers tend to travel at their desired speed in each road section but the environment (i.e. preceding vehicle, adjacent vehicles, traffic signals, signs, blockages, etc.) conditions their behaviour. Simulation time is split into small time intervals called simulation cycles or simulation steps (Δt).

At each simulation step, the position and speed of every vehicle in the system is updated according to the following algorithm:

```
if (it is necessary to change lanes) then
    Apply Lane-Changing Model
endif
if (the vehicle has not changed lanes) then
    Apply Car-Following Model
endif
```

Once all vehicles have been updated for the current simulation step, vehicles scheduled to arrive during this cycle are introduced into the system and the next vehicle arrival times are generated.

5.3.3.1 Microscopic Car Following

The car-following model implemented in Aimsun is based on the model proposed by Gipps (1981, 1986b). It can actually be considered an evolution of this empirical model, in which the model parameters are not global but determined by the influence of local parameters depending on the type of driver (speed limit acceptance of the vehicle), the road characteristics (speed limit on the section, speed limits on turnings, etc.), the influence of vehicles on adjacent lanes, etc.

The model consists of two components: acceleration and deceleration. The first represents the intention of a vehicle to achieve a certain desired speed, while the second reproduces the limitations imposed by the preceding vehicle when trying to drive at the desired speed.

This model states that the maximum speed to which a vehicle (n) can accelerate during a time period (t, $t+T$) is given as

$$V_{\mathrm{a}}(n,t+T) = V(n,t) + 2.5a(n)T\left(1 - \frac{V(n,t)}{V^{*}(n)}\right)\sqrt{0.025 + \frac{V(n,t)}{V^{*}(n)}} \tag{5.1}$$

where:

$V(n, t)$ is the speed of the vehicle n at time t;
$V^{*}(n)$ is the desired speed of the vehicle (n) for current position;
$a(n)$ is the maximum acceleration for the vehicle n;
T is the reaction time.

On the other hand, the maximum speed that the same vehicle (n) can reach during the same time interval (t, $t+T$), according to its own characteristics and the limitations imposed by the presence of the lead vehicle ($n-1$), is

$$V_b(n,t{+}T) = d(n)T + \sqrt{d(n)^2T^2 - d(n)\left[2\left\{x(n-1,t) - s(n-1) - x(n,t)\right\} - V(n,t)T - \frac{V(n-1,t)^2}{d'(n-1)}\right]} \tag{5.2}$$

where

$d(n)$ (< 0) is the maximum deceleration desired by vehicle n;
$x(n, t)$ is the position of the vehicle n at time t;
$x(n-1, t)$ is the position of the preceding vehicle $(n-1)$ at time t;
$s(n-1)$ is the effective length of the vehicle $(n-1)$;
$d'(n-1)$ is an estimation of the vehicle $(n-1)$ desired deceleration.

The speed of the vehicle (n) during time interval $(t, t{+}T)$ is the minimum of the two expressions above:

$$V(n,t+T) = \min\ \{\ V_{\mathrm{a}}(n,t+T), V_{\mathrm{b}}(n,t+T)\ \} \tag{5.3}$$

Then, the position of the vehicle n inside the current lane is updated taking this speed into the movement equation:

$$x(n,t+T) = x(n,t) + V(n,t+T)T \tag{5.4}$$

The car-following model is such that a leading vehicle, i.e. a vehicle driving freely without any vehicle affecting its behaviour, would try to drive at its maximum desired speed. Three parameters are used to calculate the maximum desired speed of a vehicle while driving on a particular section or turning; of those, two are related to the vehicle and one to the section or turning. Specifically

1. Maximum desired speed of the vehicle i: $v_{\max}(i)$
2. Speed acceptance of the vehicle i: $\theta(i)$
3. Speed limit of the section or turning s: $S_{\text{limit}}(s)$

The speed limit for a vehicle i on a section or a turning s, $s_{\text{limit}}(i,s)$, is calculated as follows:

$$s_{\text{limit}}(i,s) = S_{\text{limit}}(s) \cdot \theta(i) \tag{5.5}$$

Then, the maximum desired speed of the vehicle i on a section or a turning s, $v_{\max}(i,s)$, is calculated as follows:

$$v_{\max}(i,s) = MIN\left[s_{\text{limit}}(i,s), v_{\max}(i)\right] \tag{5.6}$$

This maximum desired speed $v_{\max}(i,s)$ is the same as that referred to above, in the Gipps' car-following model, as $V^*(n)$ [see eq. (5.1)].

The car-following model proposed by Gipps is a one-dimensional model that considers only the vehicle and its leader. However, the implementation of the car-following model in Aimsun also considers the influence of adjacent lanes. When a vehicle is driving along a section, we consider the influence that a certain number of vehicles driving slower in the adjacent right-side lane – or left-side lane when driving on the left – may have on the vehicle. The model determines a new maximum desired speed of a vehicle in the section, which will then be used in the car-following model, considering the mean speed of vehicles driving downstream of the vehicle in the adjacent slower lane and allowing a maximum difference of speed.

5.3.3.2 Microscopic Lane-Changing Model

The lane-changing model can also be considered as a development of the lane-changing model proposed by Gipps (1986a, b). Lane change is modelled as a decision process, analysing

- the desirability or necessity of a lane change (such as for imminent turning manoeuvres determined by the overall route that the vehicle is following);
- the benefits of a lane change (for example to reach the desired speed when the leading vehicle is slower); and
- the feasibility conditions for a lane change that are also local, depending on the location of the vehicle in the road network.

The lane-changing model is a decision model that approximates the driver's behaviour in the following manner: each time a vehicle has to be updated, we ask the following question*: Is it necessary or desirable to change lanes?* The answer to this question will depend on the distance to the next turning and the traffic conditions in the current lane. The traffic conditions are measured in terms of speed and queue lengths. When a driver is going slower than he wishes, he tries to overtake the preceding vehicle. On the other hand, when he is travelling fast enough, he tends to go back into the slower lane.

If we answer the previous question in affirmatively, to successfully change lanes, we must first answer two further questions:

- Is there benefit to changing lane? Check whether there will be any improvement in the traffic conditions for the driver as a result of lane changing. This improvement is measured in terms of speed and distance. If the speed in the future lane is fast enough compared to the current lane, or if the queue is short enough, then it is beneficial to change lanes.
- Is it feasible to change lanes? Verify that there is enough of a gap to make the lane change with complete safety. For this purpose, we calculate both the braking imposed by the future downstream vehicle to the changing vehicle and the braking imposed by the changing vehicle to the future upstream vehicle. If both braking ratios are acceptable, then the lane change is possible.

In order to achieve a more accurate representation of the driver's behaviour in the lane-changing decision process, three different zones inside a section are considered, each one corresponding to a different lane-changing motivation. These zones are characterized by the distance up to the end of the section, i.e., the next point of turning (see Fig. 5.2).

- *Zone 1*: This is the furthest distance from the next turning point. The lane-changing decisions are mainly governed by the traffic conditions of the lanes involved. The necessity of a future turning movement is not yet taken into account. To measure the improvement that the driver will get from changing lanes, we consider several parameters: desired speed of driver, speed and distance of current preceding vehicle, speed and distance of future preceding vehicle in the destination lane.
- *Zone 2*: This is the intermediate zone. It is mainly the desired turning lane that affects the lane-changing decision. Vehicles not driving in valid lanes (i.e. lanes where the desired turning movement can be made) tend to get closer to the correct side of the road from which the turn is allowed. Vehicles looking for a gap may try to adapt to it but do not yet affect the behaviour of vehicles in the adjacent lanes.
- *Zone 3*: This is the shortest distance to the next turning point. Vehicles are forced to reach their desired turning lanes, reducing speed if necessary, and even come to a complete stop (*gap forcing*) in order to make the change possible. Within this zone, vehicles in the adjacent lane may also modify their behaviour (*courtesy yielding*) in order to provide a gap big enough for the vehicle to succeed in changing lanes.

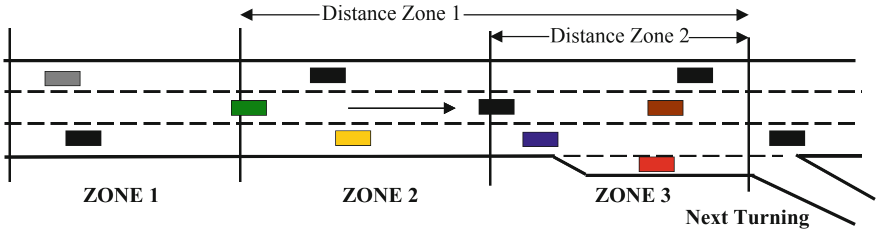


Fig. 5.2 Lane-changing zones

An overview of the lane-changing model is displayed in Fig. 5.3. The system identifies the type of entity (central lane, off-ramp lane, junction, on-ramp, etc.) into which the manoeuvre is to be carried out and then determines how zone modelling should be applied. The current traffic conditions are analysed, the level at which the lane change can be performed is determined and then the corresponding model is applied.

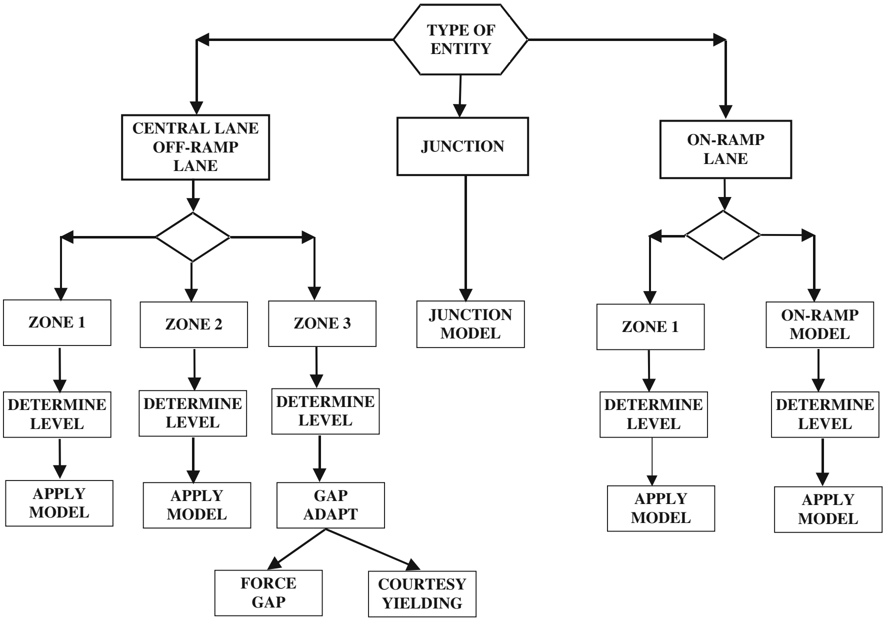


Fig. 5.3 Lane-changing model logic

5.3.3.3 Microscopic Look-Ahead Model

Until now, we have made the assumption that a vehicle driving along a section has knowledge only of its next turning movement, that is the turning it will take when arriving at the end of the current section. This means that the lane-changing decisions of each particular vehicle are made according to the next turning movement at the next intersection. However, this approach is not safe in urban networks with short sections or motorways/freeways with weaving sections that may be relatively short. In such situations and with very heavy traffic congestion, if we take into account only the next turning movement in the lane-changing decisions, it is possible that some vehicles will not reach the appropriate turning lane and consequently miss the next turn.

In order to avoid this undesirable behaviour as much as possible, Aimsun includes a look-ahead model, whose main purpose is to make the vehicles able to reach the turning lane in time. The idea is to provide vehicles with the knowledge of a set of next turning movements (defined by the user) based on which they can make lane-changing decisions.

The look-ahead model comprises four steps:

1. Knowing all sections belonging to the path that a vehicle follows, it considers the next turning movements ahead in the lane-changing behaviour. The lane-changing decisions are influenced by close lanes, the reserved lanes, bus stops,

in case of public transport vehicles, of all sections in its path and finally by the tuning movements ahead.
2. Lane-changing zones 2 and 3 of any section are extended back beyond the limits of the section, therefore affecting the upstream sections.
3. The next turning movement also influences the turning manoeuvres, so the selection of destination lane is also made based on the next turn.
4. Greater variability is given to the lane-changing zones in order to distribute the lane-changing manoeuvres along a longer distance.

5.3.3.4 Microscopic Gap-Acceptance Model

A gap-acceptance model is used to model give-way behaviour. This model determines whether a lower priority vehicle approaching an intersection can or cannot cross depending on the circumstances of higher priority vehicles (position and speed). This model takes into account the distance of vehicles to the hypothetical collision point, their speeds and their acceleration rates. It then determines the time needed by the vehicles to clear the intersection and produces a decision that is also a function of the level of risk of each driver.

Several vehicle parameters may influence the behaviour of the gap-acceptance model: acceleration rate, desired speed, speed acceptance and maximum give-way time. Other parameters, such as visibility distance at the intersection and turning speed, which are related to the section, may also have an effect. Among these, the acceleration rate, the maximum give-way time and the visibility distance at the intersection are the most important.

The acceleration rate gives the acceleration capability of the vehicle and therefore has a direct influence on the required safety gap. The maximum give-way time is used to determine when a driver starts to get impatient if he/she cannot identify a gap. When the driver has been waiting for more than this time, the safety margin (normally two simulation steps) is reduced to half of it (only one step).

The following algorithm is applied in order to determine whether a vehicle approaching a give-way sign can cross or not:

Given a vehicle (VEHY) approaching a give-way junction,

1. Obtain the closest higher priority vehicle (VEHP)
2. Determine the theoretical collision point (TCP)
3. Calculate time (TP1) needed by VEHY to reach TCP
4. Calculate estimated time (ETP1) needed by VEHP to reach TCP
5. Calculate time (TP2) needed by VEHY to cross TCP
6. Calculate estimated time (ETP2) needed by VEHP to clear the junction
7. If TP2 (plus a safety margin) is less than ETP1, vehicle VEHY will have enough time to cross; therefore it will accelerate and cross
8. Otherwise, if ETP2 (plus a safety margin) is less than TP1, vehicle VEHP would have already crossed TCP when VEHY had reached it; searching for the next closest vehicle with a higher priority, it would become VEHP and go to step 2
9. Else, vehicle VEHY must give way, decelerating and stopping if necessary

5.3.4 Modelling Mesoscopic Vehicle Movement

Mesoscopic vehicle movement in Aimsun is modelled depending on the location of a vehicle:

- Modelling vehicle movement in sections: car following and lane changing
- Modelling vehicle movement in nodes (node model):
  - Modelling vehicle movement in turnings
  - Modelling vehicle movement from sections to turnings: apply gap-acceptance model
  - Modelling vehicle movement from turnings to sections: apply lane selection model

Figure 5.4 illustrates mesoscopic vehicle movements in Aimsun.

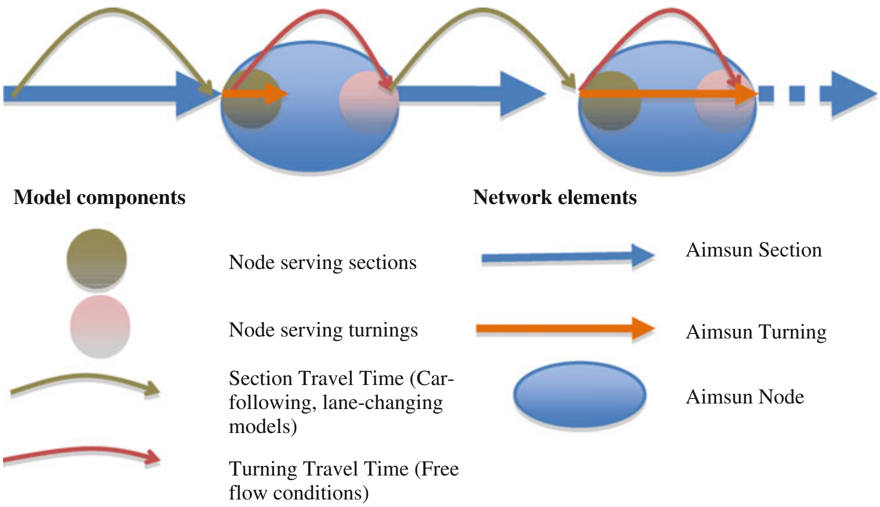


Fig. 5.4 Modelling vehicle movements

5.3.4.1 Mesoscopic Car following

In Aimsun, vehicles are assumed to move through sections and turnings, so sections and turnings are vehicle containers. That means that yellow boxes are not considered explicitly by the model. The section capacity, in terms of the number of vehicles that can stay at the same time in a section, is calculated by using the jam density (user-defined parameter) multiplied by the section length and the number of lanes. By contrast, the turning capacity is calculated in a similar way, but using the feasible connections in the node instead of the number of lanes.

Car-following and lane-changing models are applied to calculate the section travel time. This is the earliest time a vehicle can reach the end of the section, taking into account the current status of the section (number of vehicles in the section). The modelling of vehicle movements inside sections in the Aimsun mesoscopic simulator is based on the work of Mahut (1999a, b, 2001).

Aimsun mesoscopic simulator is based on this node model that moves vehicles from one section to its next section of its path. This model contains two actions that take place in all nodes:

- *Serving sections*. This server calculates the next vehicle to enter the node. This is done by applying the gap-acceptance model and then using the exit times calculated by the car-following and lane-changing models mentioned above.
- *Serving turnings*. This server calculates the next vehicle to leave the node. The selection is done by applying the car-following and lane-changing models to calculate the travel time and getting the earliest time when a vehicle can enter in its downstream section.

5.3.4.2 Mesoscopic Lane Selection Model

This model is used to calculate the origin and destination lanes. These calculations are made during the treatment of the event called 'node event from turning', that is, before a vehicle enters into a section.

The mesoscopic simulator in Aimsun calculates a default movement from all exit lanes at the beginning of the simulation. This means that from each lane in a section there is a default next lane for each turning. Besides this default next lane choice, Aimsun mesoscopic is using two more heuristics in order to decide the next lane movement: obtaining the status of the next section and employing a look-ahead model.

From the default lane choice, the mesoscopic simulator looks for the best entrance lane from all turning destination lanes. By contrast, in comparable settings, the microscopic simulator looks only for a subset of all destination lanes.

In order to decide to change the default lane choice, Aimsun takes into account the density of all lanes and the cost of changing the default lane choice.

The other way to change the default lane choice is by applying the look-ahead model described early. The look-ahead model implemented in Aimsun is shared between the microscopic and mesoscopic simulators.

5.3.4.3 Mesoscopic Gap-Acceptance Model

The gap-acceptance model is used to model give-way behaviour. In particular, the model is used when resolving node events in order to decide which of two vehicles in a conflict movement has priority. The generic rule is a FIFO rule, except when there is a traffic sign (such as stop or give-way signs) in which case a simplification of the gap-acceptance model described earlier is applied.

5.4 Dynamic Traffic Assignment

Traffic assignment is the process of determining how the traffic demand, usually defined in terms of an origin–destination matrix, is loaded onto a road network, providing the way to compute the traffic flows on the network links. The underlying hypothesis is that vehicles travel from origin to destinations in the network along the available routes connecting them. The characteristics of a traffic assignment procedure are determined by the hypothesis on how vehicles use the routes. The main modelling hypothesis is the concept of user equilibrium which states that vehicles try to minimize their individual travel times, that is, drivers choose the routes that they perceive as the shortest under the prevailing traffic conditions. This modelling hypothesis is formulated in terms of Wardrop's first principle: *The journey times on all the routes actually used are equal and less than those which would be experienced by a single vehicle on any unused route.*

Traffic assignment models based on this principle are known as user equilibrium models as opposed to models in which the objective is to optimize the total system travel time independently of individual preferences – for details, see either Sheffi (1985) or Florian and Hearn (1995).

The advent of intelligent transport systems (ITS) and more specifically advance traffic management systems (ATMS) and advanced traffic information systems (ATIS) has highlighted the requirement for models accounting for flow changes over time, that is dynamic models able to describe appropriately the time dependencies of traffic demand and the resulting traffic flows. The dynamic traffic assignment problem can thus be considered an extension of the traffic assignment problem described above, capable of describing how traffic flow patterns evolve in time and space on the network (Mahmassani, 2001). The approaches proposed to solve the DTA problem can be broadly classified into two classes: mathematical formulations looking for analytical solutions and simulation looking for approximate heuristic solutions. General simulation-based approaches (Tong and Wong, 2000; Lo and Szeto, 2002; Varia and Dhingra, 2004; Liu et al., 2005) explicitly or implicitly split the process into two parts: a route choice mechanism determining how the time-dependent flows are assigned onto the available paths at each time step and method to determine how these flows propagate in the network. A systematic approach based on these two components was proposed by Florian et al. (2001, 2002) (see Fig. 5.5).

Solving the DTA in Aimsun involves the conceptual diagram depicted in Fig. 5.5. In terms of the software, when the user selects the dynamic scenario dialog in Aimsun, the system offers two options – microscopic or mesoscopic, determining the simulation approach on which the network loading is based – and for each one two alternatives: DTA based on route choice models (Ben-Akiva and Bierlaire, 1999) or DTA based on DUE.

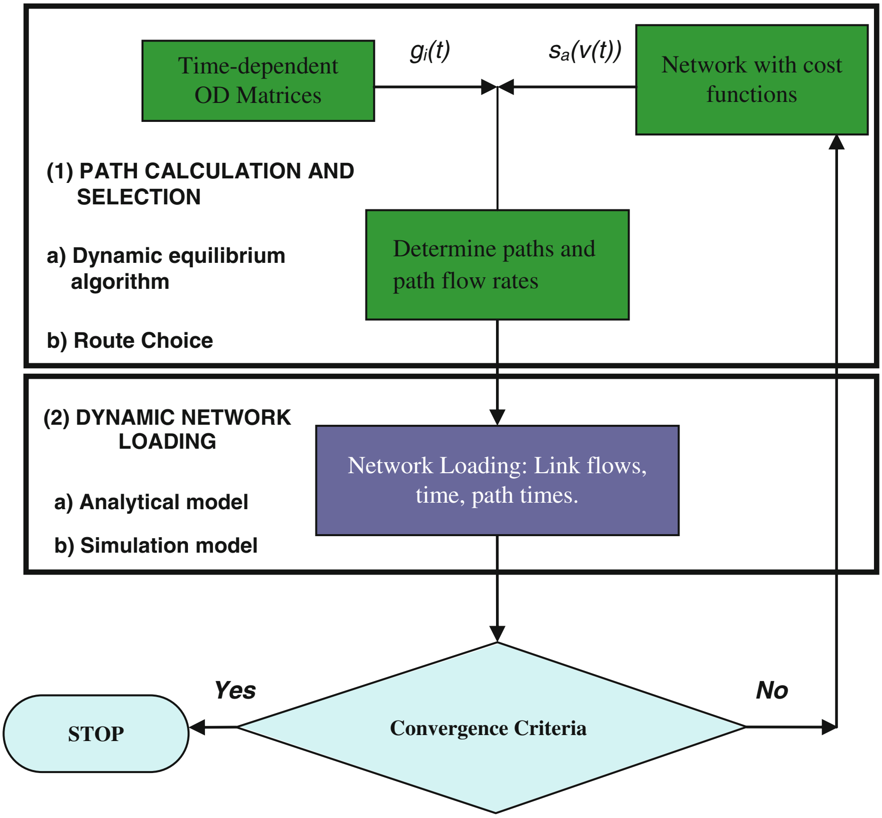


Fig. 5.5 Conceptual diagram of the heuristic dynamic assignment

The convergence criterion depends on the selected alternative: the completion of the demand loading in a one-off (single iteration) DTA based on route choice models and either the completion of the number of defined iterations or when the Rgap function reaches the desired accuracy in the DTA based on DUE.

An efficient computational implementation of this conceptual approach requires that the analytical part of the process, that is the path calculation and selection, is implemented independently of the dynamic network loading process selected to implement the heuristic part of the dynamic traffic assignment. In other words, assuming network consistency between the mesoscopic and microscopic representations, the path calculation based on time-dependent link costs must be the same; the only difference will lie in the values of the arguments of the link cost functions output by the mesoscopic or the microscopic traffic simulation.

Aimsun uses a common network representation, object model and database accessible by all models. In addition, both the microscopic and mesoscopic models are based on individual vehicles. This makes it possible to implement a 'dynamic traffic assignment server' (Barceló and Casas, 2006), whose conceptual structure

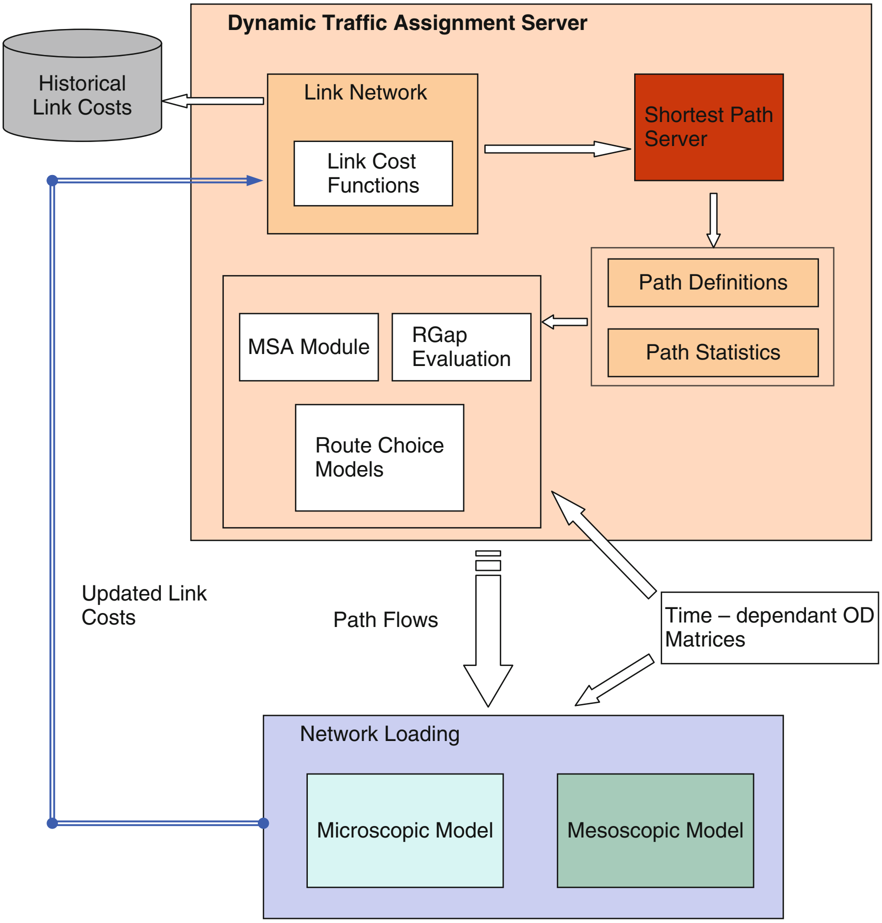


Fig. 5.6 Dynamic traffic assignment server structure

is depicted in Fig. 5.6. The dynamic traffic assignment server computes common shortest paths based on link cost functions evaluated in terms of current link costs or link cost accounting for stored values from previous iterations. Link costs are then updated by either a mesoscopic or a microscopic simulation approach. In what follows we describe three dynamic traffic assignment schemes that may be employed in a modelling study with Aimsun depending on the study objectives.

5.4.1 Dynamic Traffic Assignment Based on Discrete Choice Theory (Stochastic Route Choice)

Certain real-world situations, for example the reaction of drivers to a non-recurring incident, create a requirement for dynamic traffic assignment mechanisms whose output is not necessarily optimal in the sense of the dynamic version of Wardrop's principle (Friesz et al., 1993; Smith, 1993; Ran and Boyce, 1996). In such cases the route choice mechanism tries to optimize route selection decisions based on the currently available information. One such mechanism that accounts for uncertainties in the information available to drivers is the use of choice that includes the possibility of en route rerouting mechanisms, based either on discrete choice theory or on other probabilistic approaches (Mahmassani, 2001). These approaches provide solutions to the dynamic traffic assignment problem that does not seek dynamic user equilibrium (DUE).

One of the DTA methods in Aimsun is based on stochastic route choice (Barceló and Casas, 2002, 2004a, b, 2006).

The simulation process based on time-dependent routes consists of the following steps:

Repeat until all the demand has been assigned:

1. Calculate initial shortest routes for each OD pair using the defined initial costs.
2. Simulate for a user-defined time interval assigning to the available routes the fraction of the trips between each OD pair for that time interval according to the selected route choice model and obtaining new average link travel times as a result of the simulation.
3. Recalculate shortest routes, taking into account the current average link travel times.
4. If there are guided vehicles, or variable message signs suggesting rerouting, provide the information calculated in step 3 to the drivers that are dynamically allowed to reroute.
5. Go to step 2.

In the proposed network loading mechanism based on microscopic or mesoscopic simulation, vehicles follow paths from their origins in the network to their destinations. So the first step in the simulation process is to assign a path to each vehicle when it enters the network, from its origin to its destination. This assignment, made by a path selection process based on a discrete route choice model, will determine the path flow rates.

Given a finite set of alternative paths, the path selection calculates the probability of each available path and then the driver's decision is modelled by randomly selecting an alternative path according to the probabilities assigned to each alternative. Route choice functions represent implicitly a model of user behaviour that emulates the most likely criteria employed by drivers to decide between alternative routes in terms of the user's perceived utility (or, more precisely, a disutility or cost in the case of trip decisions) defined in terms of perceived travel times, route lengths, expected traffic conditions along the route, etc.

The logit, c-logit and proportional route choice functions are the default route choice functions available in Aimsun. Additional choice functions may be introduced by the user, using Aimsun's function editor.

The multinomial logit route choice model defines the choice probability P_k of alternative path k, $k \in K_i$, as a function of the difference between the measured utilities of that path and all other alternative paths:

$$P_k = \frac{e^{\theta V_k}}{\sum_{j \in K_i} e^{\theta V_j}} = \frac{1}{1 + \sum_{j \neq k} e^{\theta (V_j - V_k)}} \tag{5.7}$$

where V_j is the perceived utility for alternative path j (i.e. the opposite of the path cost or path travel time) and θ is a scale factor that plays a twofold role: on the one hand, it makes the decision based on differences between utilities independent of measurement units on the other hand, it influences the standard error of the distribution of expected utilities, determining in that way a trend towards utilizing many alternative routes or concentrating on very few routes. In that sense, θ is the critical parameter to calibrate so that the logit route choice model leads to a meaningful selection of routes.

A drawback in using the logit function is a 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 behaviour: 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.

The instability of the routes used can be substantially improved when the network topology allows for alternative routes with little or no overlapping at all, 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 origins and destinations exist, and some of them exhibit a certain degree of overlapping, the use of the logit function may still exhibit some weaknesses. To avoid this drawback, the c-logit model, proposed in Cascetta et al. (1996) and Ben-Akiva and Bierlaire (1999), is also available as an option. In this model, the choice probability P_k, of each alternative path k belonging to the set K_i of available paths connecting the ith OD pair, is defined as

$$P_k = \frac{e^{\theta (V_k - \mathrm{CF}_k)}}{\sum_{j \in K_i} e^{\theta (V_j - \mathrm{CF}_j)}} \tag{5.8}$$

where V_j is the perceived utility for alternative path j, i.e. the opposite of the path cost, 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 overlapping paths have a larger CF factor and therefore smaller utility compared to paths with a similar perceived utility and less overlapping. CF_k is calculated as follows:

$$\mathrm{CF}_k = \beta \cdot \ln \sum_{j \in K_i} \left(\frac{L_{jk}}{L_j^{1/2} L_k^{1/2}} \right)^{\gamma} \tag{5.9}$$

where L_{jk} is the length of arcs common to paths j and k, while L_j and L_k are the length of paths j and k, respectively. Depending on the two factor parameters β and γ, a greater or a 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_j; γ is a positive parameter, whose influence is smaller than β and which has the opposite effect. The utility V_j used in this model for path j is the opposite of the path travel time tt_j (or path cost depending on how it has been defined by the user).

Another option is the estimation of the choice probability P_k of path k, $k \in K_i$, in terms of a generalization of Kirchhoff's laws given by the function

$$P_k = \frac{\mathrm{CP}_k^{-\alpha}}{\sum_{j \in K_i} \mathrm{CP}_j^{-\alpha}} \tag{5.10}$$

where CP_j is the cost of the path j, α is in this case the parameter whose value has to be calibrated.

5.4.2 *Dynamic Traffic Assignment via an Iterative Heuristic (Stochastic Route Choice with Memory/Additional Information)*

The formulation described in this section is useful for scenarios in which ATMS and ATIS applications transmit reliable (simulation-based) traffic forecasts to drivers, allowing them to adjust their behaviour accordingly. Alternatively it may be considered as a model of the process by which travellers adjust their current information with conjectures about the expected traffic conditions ahead; this could correspond to the process followed by commuters adapting their behaviour according to a day-to-day learning process depending on the fluctuations of traffic patterns until they consider that no further improvement is possible. The implementation in Aimsun combines dynamic (mesoscopic or microscopic) simulations with an iterative heuristic procedure that mimics the day-to-day learning process that attempts to reach DUE, though with no guarantee of convergence (Barceló and Casas, 2002, 2006; Liu et al., 2005).

The iterative heuristics replicates the simulation N times and link costs for each link j, for each time interval t, $t+1$, ..., L (where $L = T/\Delta t$, T being the simulation horizon and Δt the user-defined time interval in which to update paths and path flows) at every iteration n stored. Thus at iteration n, the link costs of previous iteration $n-1$ can be used as an anticipatory mechanism to estimate the expected link cost at the current iteration. Let $s_a^{jl}(v)$ be the current cost of link a with flow v 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 given as

$$\bar{s}_a^{j,l+i}(v) = \frac{1}{j-1}\sum_{m=1}^{j-1} s_a^{m,l+i}(v); i = 1, ..., L-l \tag{5.11}$$

The ‘forecasted’ link cost can then be computed as

$$\tilde{s}_a^{j,l+1}(v) = \sum_{i=0}^{L-l} \alpha_i \bar{s}_a^{j,l+i}(v); \quad \text{where} \sum_{i=0}^{L-l} \alpha_i = 1,\ \alpha_i \geq 0,\ \forall i \text{ are weighting factors} \tag{5.12}$$

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

$$\tilde{S}_k(h^{l+1}) = \sum_{a \in A} \tilde{s}_a^{j,l+1}(v)\,\delta_{ak} \tag{5.13}$$

where, usually, δ_{ak}, the arc-path incidence matrix, is 1 if link a belongs to path k and 0 otherwise. The path costs $\tilde{S}_k\left(h^{l+1}\right)$ are the arguments of the route choice function (logit, c-logit, proportional or user-defined) used at iteration l+1 to distribute the demand g_i^{l+1} across the available paths for OD pair i.

The default implementation in the current version of Aimsun uses a simplified version consisting of the link cost function:

$$c_{it}^{k+1} = \lambda c_{it}^{k} + (1-\lambda)\,\tilde{c}_{it}^{k} \tag{5.14}$$

where c_{it}^{k+1} is the cost of using link i at time t at iteration k+1 and c_{it}^{k} and $\tilde{c}_{it}^{k}$ correspond to the expected and experienced link costs, respectively, at this time interval from previous iterations.

5.4.3 Dynamic Traffic Assignment via the Method of Successive Averages (Dynamic User Equilibrium)

Several road infrastructure modification programmes give rise to the problem of understanding ‘steady-state’ time-dependent path choice after the modifications have been in place for a while. In many cases, the motivation at planning level is not to understand the pattern and duration of the transition from the current state to the final state (something attempted by the iterative heuristics described in the previous section) but rather to approximate that final state assuming that Wardrop’s generalized principle applies (i.e. reaching or approaching dynamic user equilibrium).

The method of successive averages (MSA) procedure redistributes the flows among the available paths in an iterative procedure that at iteration n computes a new shortest path from origin r to destination s at time interval t, $c_{rs}(t)$; the path flow update process is as follows:

Case a $c_{rs}(t) \notin P_{rs}^{n}(t)$

$$f_{rsp}^{n+1}(t) = \begin{cases} \alpha_n f_{rsp}^{n}(t) & \text{if } p \in P_{rs}^{n}(t) \\ & \forall r, s, t \\ (1-\alpha_n)\, d_{rs}(t) & \text{if } p = c_{rs}(t) \end{cases}$$

Let $P_{rs}^{n+1}(t) = P_{rs}^{n}(t) \cup c_{rs}(t)$
Case b $c_{rs}(t) \in P_{rs}^{n}(t)$

$$f_{rsp}^{n+1}(t) = \begin{cases} \alpha_n f_{rsp}^{n}(t), & \text{if } p \neq c_{rs}(t) \\ & \forall r, s, t \\ \alpha_n f_{rsp}^{n+1}(t) + (1-\alpha_n)\, d_{rs}(t), & \text{if } p = c_{rs}(t) \end{cases}$$

Let $P_{rs}^{n+1}(t) = P_{rs}^{n}(t)$
where $f_{rsp}(t)$ is the flow on path p from r to s departing origin r at time interval t, $P_{rs}(t)$ is the set of all available paths from r to s at time interval t and $d_{rs(t)}$ is the demand (number of trips) from r to s at time interval t.

Depending on the values of the weighting coefficients α_n, different MSA schemes can be implemented (Carey and Ge, 2007), perhaps the most typical value is $\alpha_n = n/(n+1)$.

One of the potential computational drawbacks of these implementations of MSA is the growing number of paths in the case of large networks; to avoid this in the case of the DTA assignments in Aimsun, the user has the option to specify the maximum number K of paths to keep for each origin–destination pair. Therefore in implementing the MSA in Aimsun, it was considered that it would be desirable to keep this feature.

Several modified MSA implementations have been proposed to keep control on the number of paths in MSA algorithms (Peeta and Mahmassani, 1995; Sbayti et al., 2007); however, possibly one of the most efficient computationally is the one proposed by Florian et al. (2002) modifying the MSA algorithm to keep bounded the number of alternative paths to account for each origin–destination pair. This variant of the algorithm initializes the process on the basis of an incremental loading scheme distributing the demand among the available shortest paths; the process is repeated for a predetermined number of iterations after which no new paths are added and the corresponding fraction of the demand is redistributed according to the MSA scheme. The modified MSA works as follows:

Let K be the maximum number of iterations to compute new paths.
Case a: $n \leq K$: a new shortest path $c_{rs}(t) \notin P_{rs}^{n}(t)$ is found

$$f_{rsp}^{n+1}(t) = \frac{1}{n+1} d_{rs}(t),\ \forall p \in P_{rs}^{n}(t), \forall (r,s) \in \Im, \forall t \in [0, T]$$

Let $P_{rs}^{n+1}(t) = P_{rs}^{n}(t) \cup c_{rs}(t)$

Case b: $n > K$: the new shortest path is computed among the existing paths $c_{rs}(t) \in P_{rs}$, and the set P_{rs} does not change

$$f_{rsp}^{n+1}(t) = \begin{cases} \frac{n}{n+1} f_{rsp}^{n}(t) & \text{if } p \neq c_{rs}(t) \\ \frac{n}{n+1} f_{rsp}^{n+1}(t) + \frac{1}{n+1} d_{rs}(t) & \text{if } p = c_{rs}(t) \end{cases} \quad \forall p \in P_{rs}, \forall (r,s) \in \Im, \forall t \in [0,T]$$

This is the version of the MSA algorithm implemented in Aimsun. However, taking into account the possibility of repeating shortest paths from one iteration to next to keep a maximum of K different shortest paths, a proper implementation of the algorithm requires that the number of iterations n is defined by OD pair and time interval.

All the proposed approaches for DUE are based on simulation procedures for the network loading process and therefore are heuristic in nature; therefore no formal proof of convergence can be provided, and consequently a way of empirically determining whether the solution reached can be interpreted in terms of a DUE, in the sense that 'the actual travel time experienced by travellers departing at the same time are equal and minimal', can be based on an ad hoc version of the relative gap function proposed by Janson (1991):

$$\mathrm{Rgap}(n) = \frac{\sum_{t} \sum_{(r,s)\in\Im} \sum_{p\in P_{rs}(t)} f_{rsp}^{n}(t) \left[\tau_{rsp}^{n} - \theta_{rs}^{n}(t)\right]}{\sum_{t} \sum_{(r,s)\in\Im} d_{rs}(t)\, \theta_{rs}^{n}(t)} \tag{5.15}$$

where $f_{rsp}^{n}(t)$ is the flow on path p from r to s departing origin r at time t at iteration n and the difference $\tau_{rsp}^{n}(t) - \theta_{rs}^{n}(t)$ measures the excess cost experienced by the fact of using a path of cost $\tau_{rsp}^{n}(t)$ instead of the shortest path of cost $\theta_{rs}^{n}(t)$ at iteration n. The ratio measures the total excess cost with respect to the total minimum cost if all travellers had used shortest paths.

5.4.4 Methodology and Data Flows for Dynamic Traffic Assignment

The introductory section has highlighted the advantages of full integration whereby all models share a network representation, modelling object data and demand data. From a practitioner's standpoint what is crucially important is the ability to combine model outputs, thus giving rise to sophisticated workflows where models are applied sequentially, iteratively or even concurrently (Barceló and Casas, 2006). Figure 5.7 illustrates the possibilities that open up in terms of workflows.

We distinguish two main sets of data flows:

- OD matrix data flows
- Path assignment data flows

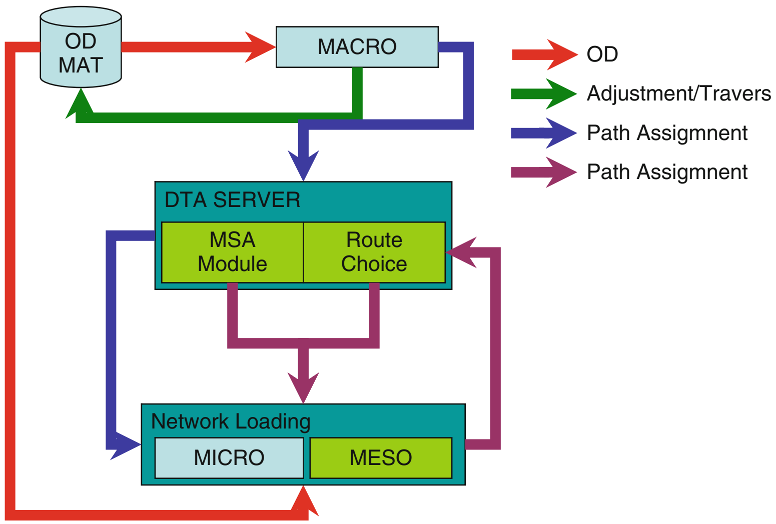


Fig. 5.7 Macro–meso–micro data flows

5.4.4.1 OD Matrix Data Flows

- To refine the inputs for the microscopic/mesoscopic simulation:
  - Estimating demand for future scenarios by means of growing factor analysis and matrix-balancing procedures.
  - Adjusting the global OD matrix from the available traffic counts on a subset of links.
- To start the analysis at the macrolevel for a large urban or metropolitan area and refine the analysis at the micro/mesolevel conducting detailed dynamic simulation experiments of selected subareas:
  - Defining interactively the window spanning the selected subarea
  - Calculating the corresponding traversal matrix
  - Adjusting the traversal from traffic counts of links in the network spanning the subarea
  - Running the simulation experiments for the subarea model

5.4.4.2 Path Assignment Data Flows

The main path assignment data flows considering the functional architecture and the integration of macro–meso–micro could be summarized as follows:

- Path assignment results could be the output of the following:
  - *Static traffic assignment applying the macroscopic model*: The path definition and the path flow rates are per the whole simulation time (no per time period).

  The result, as a consequence of the static user equilibrium, could be interpreted as a reflection of a recurrent situation or historical situation without the concept of time.
- *Dynamic traffic assignment applying the DUE approach with independency of the type of network loading process (either could be mesoscopic or microscopic model)*: In that case, the path definition and the path flow rates are per time interval (defined by the user). The result, as a consequence of the dynamic user equilibrium, could be interpreted as a reflection of a recurrent situation over the time, giving the 'normal' traffic behaviour or profile, considering the time dimension.

• Path assignment results could be the input of the following:

  - *Dynamic traffic assignment applying the dynamic user equilibrium approach*: The path assignment result, concretely the path definition, could be used as initial paths, instead to start with the paths calculated in free-flow conditions, per time slice if the path assignment result comes from a dynamic traffic assignment based on DUE approach using either mesoscopic or microscopic model as network loading process or per the whole simulation if the path assignment result comes from a static traffic assignment using the macroscopic model.
  - *Dynamic traffic assignment based on discrete route choice models*: The path assignment result of either a static traffic assignment or a dynamic traffic assignment based on DUE, interpreted as the recurrent situation, could be used to define the paths and the path flow rates for a subset of the drivers (for instance, defining a certain percentage of use of the path assignment result) and the rest of the vehicles decide the path according to the dynamic traffic assignment based on discrete route choice models.

The different alternatives of generating the path assignment results depending on the type of traffic assignment (either static or dynamic) could be interpreted:

1. the user equilibrium interpreted as the result of the recurrent or day-to-day learning processes of the drivers, either dynamic or static, and
2. the dynamic traffic assignment based on discrete route choice models interpreted as the vehicles receive information about the current situation.

We can consider different applications of the path assignment result data flow in projects:

• Model the addition of a new infrastructure or the modification of a current infrastructure:

  - One possibility is the evaluation of the scenario base (the current situation) and the future scenario, applying the dynamic traffic assignment based on dynamic user equilibrium in order to model the recurrent situation in both scenarios.

  - The other possibility is the evaluation of the previous scenarios, but including an intermediate scenario (transitory scenario) that represents the temporal modifications of the infrastructure due to the road works (such as lane closures, speed reductions, rerouting actions). In this intermediate scenario the recurrent situation probably is not representative, because there is no day-to-day learning process and the behaviour of the drivers could be helped by traffic management policies. In that case, the most appropriate approach is to combine the day-to-day learning process of the current situation, which means the use of the path assignment result of a dynamic user equilibrium, with a dynamic traffic assignment based on discrete route choice models in order to model the transition and the effect of the temporal traffic management policies.

- Analyse microscopically or mesoscopically a subarea but considering the influence in a wide area:
  - Simulate the wide area using the dynamic traffic assignment based on dynamic user equilibrium with the mesoscopic model as network loading.
  - The path assignment result of the previous simulation could be used as input for a simulation combined with a dynamic traffic assignment based on discrete route choice models in order to model the changes of behaviour in the function of the new traffic management or the design of the subarea. The level of detail of the subarea determines the type of the network loading to consider: microscopic or mesoscopic, independently of the traffic assignment approach.

5.5 Calibration and Validation of Aimsun models

5.5.1 General Remarks

Validation is the process of testing that a model represents a viable and useful alternative means to real experimentation. This requires the exercise of *calibrating the model,* that is adjusting model parameters until the resulting output data agree closely with the system observed data. The validation of the simulation model will be established by comparing the observed output data from the actual system and the output data from the simulation experiments conducted with the computer model.

Model calibration and validation is inherently a statistical process in which the uncertainty due to data and model errors should be accounted for. Depending on the variables selected, the system and simulated data available, and their characteristics and statistical behaviour, there exists a variety of statistical techniques either for paired comparisons or for multiple comparisons and time series analysis. The conceptual framework for this validation methodology is described in the diagram of Fig. 5.8. According to this logic, when the results of the comparison analysis are not acceptable to the degree of significance defined by the analyst, the rejection of the

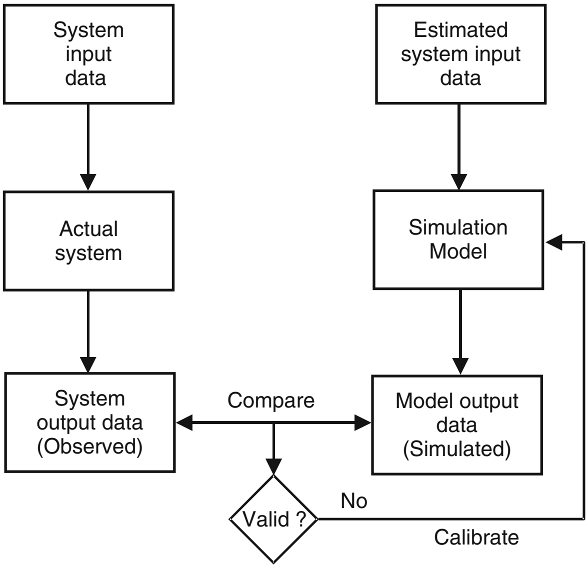


Fig. 5.8 Conceptual framework for model validation

simulation results implies the need for recalibrating some aspects of the simulation model. The process is repeated until a statistically significant degree of similitude is achieved.

In the case of traffic systems, the behaviour of the actual system is usually defined in terms of flows, speeds, occupancies, queue lengths, and so on, which can be measured by traffic detectors at specific locations in the road network. To validate the traffic simulation model, Aimsun emulates the traffic detection process and produces a series of simulated observations whose comparison to the actual measurements is used to determine whether the desired accuracy in reproducing the system behaviour is achieved. Rouphail and Sacks (2003) propose the following set of guiding principles:

1. The analyst must be aware that calibration and validation are conducted in particular contexts.
2. Depending on the context, the model requires specific sets of relevant data.
3. Both models and field data contain uncertainties.
4. Feedback is necessary for model use and development.
5. Model validation must be exercised on an independent data set from the calibration data set.

The analyst will have to identify which data are relevant for the planned study, collect them, identify the uncertainties, filter out the data accordingly and use two independent sets of data. *The first set should be used for calibrating the model parameters and the second for running the calibrated model and then for validating the calibrated model*.

The key question in the diagram of figure 5.8: "Is the model valid?" can then be reformulated as, "Do model results represent faithfully the aspect of reality that is material to the study?" The statistical techniques provide a quantified answer to this question, quantification that, according to Rouphail and Sacks, can be formally stated in the following terms: the probability that the difference between the 'reality' and the simulated output is less than a specified tolerable difference within a given level of significance:

$$P\{\ |\text{'reality'} - \text{simulated output}|\ \le d\ \} > \alpha$$

where d is the tolerable difference threshold indicating how close the model is to reality and α is the level of significance that tells the analyst how certain is the result achieved.

5.5.2 Verification and Validation in Aimsun

5.5.2.1 Verification

The main components of a traffic simulation model are the following:

1. The geometric representation of the road traffic network and related objects (traffic detectors, variable message panels, traffic lights, etc.)
2. The representation of traffic management schemes (directions of vehicle's movement, allowed and banned turnings, etc.), and of traffic control schemes (phasing, timings, offsets)
3. The individual vehicle behavioural models: car following, lane change, gap acceptance, etc.
4. The representation of the traffic demand
5. Input flow patterns at input sections to the road model and turning percentages at intersections
6. Time-sliced OD matrices for each vehicle class
7. The dynamic traffic assignment model

The graphical edition in Aimsun has been designed with the objective of supporting the user in tasks (1) and (2) of the process of building the road network model. To facilitate these tasks, Aimsun accepts as a background a digital map of the road network, in terms of a DXF file from a GIS or an AutoCAD system, a JPEG or a bitmap file, etc.; so sections and nodes can be built subsequently into the foreground. Aimsun supports both urban and interurban roads, which means that the level of detail covers elements such as surface roads, entrance and exit ramps, intersections, traffic lights and ramp metering.

The use of the graphic editors on the digital maps of the road networks provides the basis for a continuous visual validation of the quality of the geometric model. At the same time, the auxiliary online debugging tools in Aimsun prevent the most blatant mistakes in building the geometric representation, warning the modeller when obvious inconsistencies may occur.

In other words, the Aimsun model-building process is assisted with validation tools to check the correctness of the geometric model of the road network within the limits of logic rules. Some aspects may lie beyond the analysis capabilities of the assistance software, i.e. whether banning a turning is correct or not. This decision may depend on the objectives of the traffic management scheme defined by the traffic manager. Something similar could be said regarding whether or not to include a movement in a phase. However, a different case might be that of a previously defined movement that was not included in any phase; this is something that can be checked by the assistance tools.

In this way, Aimsun ensures a geometric model exhibiting a 'high face validity' that could even be further validated by the modeller through visual inspection facilitated by the graphic display of the Aimsun model.

In order to make the validation of geometry easier, Aimsun offers two tools for checking whether there are errors in the network definition or not and also gives facilities for fixing these errors.

In Fig. 5.9 an example of a network with a centroid configuration is shown. The network checker detected three problems:

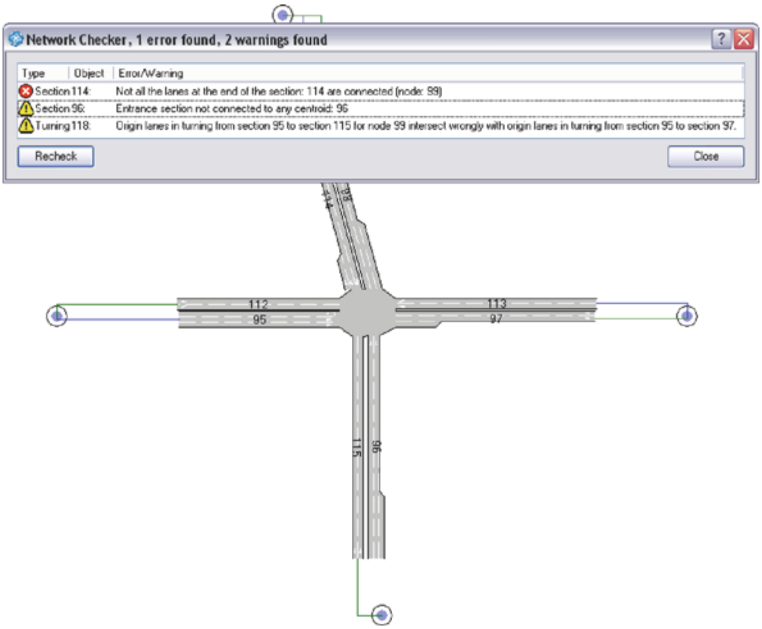


Fig. 5.9 Checking the network including centroids

- At the end of section 114, the lane on the right has no turning defined; this can be easily observed, as the node does not cover this lane.
- Section 96 is an entrance section (vehicles should enter the network through it) but it is not connected to an origin centroid generating traffic.
- There are two turnings, from section 95 to 115 and from section 95 to section 97, which would make the vehicles to collide.

Another tool available for the verification phase is the dynamic network checker (see Fig. 5.10). The purpose of this component is to detect problems within a running simulation. The specific features are the following:

- Count lost vehicles in nodes: Track all vehicles that have been unable to make their desired turning in a node.
- Stationary vehicle detection: Determine the vehicle that has been stationary for a time greater than *user-defined parameter.* Any of three actions may be taken. (i) The vehicle may be recorded in the log (either the log window or the file defined as *Output File*), (ii) the simulation may be stopped by selecting *Stop Simulation* and (iii) the vehicle may be automatically removed by selecting *Automatically Remove Vehicle*. The check may be applied only to specific sections of interest (and reduce execution time) by specifying section ids in a comma-separated list in the field *Apply to Sections*.

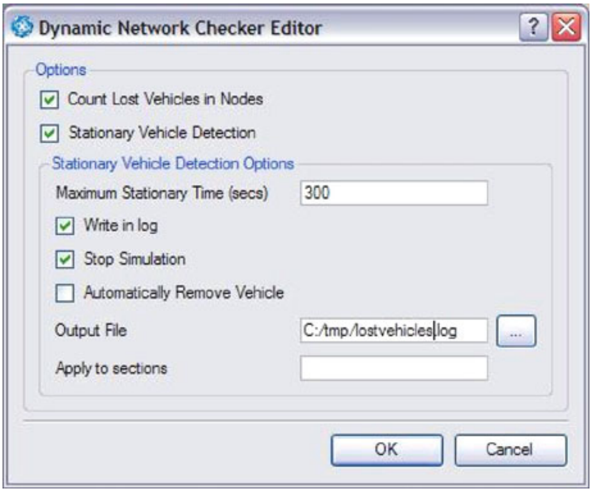


Fig. 5.10 Dynamic network checker editor

5.5.3 Validation

The statistical methods and techniques for validating simulation models are clearly explained in most textbooks and specialized papers (Law and Kelton, 1991; Balci, 1998; Kleijnen, 1995) and the validation process is independent of the traffic simulation model.

From the general methodology, three main principles that establish a framework for model validation are (Barceló and Casas, 2004b) as follows:

1. The measured data in the actual system should be split into two data sets: the data set that will be used to develop and calibrate the model and a separate data set that will be used for the validation test.
2. The data collection process is specified in the system as well as in the simulation model: traffic variables or MOEs (i.e. flows, occupancies, speeds, service levels, travel times, etc.), whose values will be collected to be used for the calibration and validation phases, and the collection frequency (i.e. 30 s, 1 min, 5 min, etc.).
3. According to the methodological diagrams in Fig. 2, validation should be considered an iterative process; at each step in the iterative validation process, a simulation experiment will be conducted. Each of these simulation experiments will be defined by the data input to the simulation model, the set of values of the model parameters that identify the experiment and the sampling interval.

The validation process based on standard statistical comparison between model and system outputs (Barceló and Casas, 2004b) could be summarized as follows:

- Comparison based on global measurements
- Comparison based on time series analysis
- Comparison based on band analysis

5.5.3.1 Comparison Based on Global Measurements

A method that has been widely used in validating transport-planning models, in the typical situation in which only aggregated values are available (i.e. flow counts at detection stations aggregated to the hour), has been to analyse the scattergram or, alternatively, to use a global indicator as the GEH index, widely used in practice in the United Kingdom. Figure 5.11 depicts an example of such analysis. The regression line of observed versus simulated flows at 76 detection stations for the aggregated 1 h values is plotted along with the 95% prediction interval. The R^2 value of 93.4 and the fact that only three points lay out of the confidence band would lead to the conclusion that the model could be accepted as significantly close to the reality.

For the same example the GEH value is 72%; therefore the model would have been rejected, leading to a conclusion contradicting the previous one.

Independent of the considerations of whether one criterion is better that the other, one should draw the attention that this type of indicators can be considered only as a primary indicator for acceptance or rejection in the case of microscopic simulation models. As indicators working with aggregated values, they do not capture what

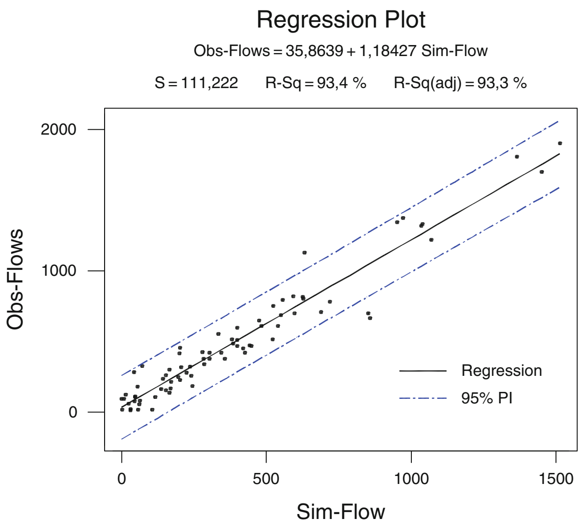


Fig. 5.11 Example of scattergram analysis to compare observed versus simulated aggregated flows

is considered the essence of the microscopic traffic simulation: the ability to capture the time variability of traffic phenomena. Therefore other types of statistical comparison should be proposed.

5.5.3.2 Comparison Based on Time Series Analysis

Theil's U-statistic (Theil, 1966) is the measure achieving the above-mentioned objectives of overcoming the drawbacks of the RMSE index and taking into account explicitly the fact that we are comparing two autocorrelated time series, and therefore the objective of the comparison is to determine how close both time series are.

An immediate interpretation of Theil's U-statistic is the following:

$U = 0 \Leftrightarrow$ the forecast is perfect
$U = 1 \Leftrightarrow$ the forecast is as bad as possible

Figure 5.12 depicts the statistics provided by Aimsun for helping in the validation of the model:

- Regression analysis
- Theil's coefficient

This information could also be displayed over the space as a global view in the network, considering the GEH index or the Theil's coefficient (see Fig. 5.13)

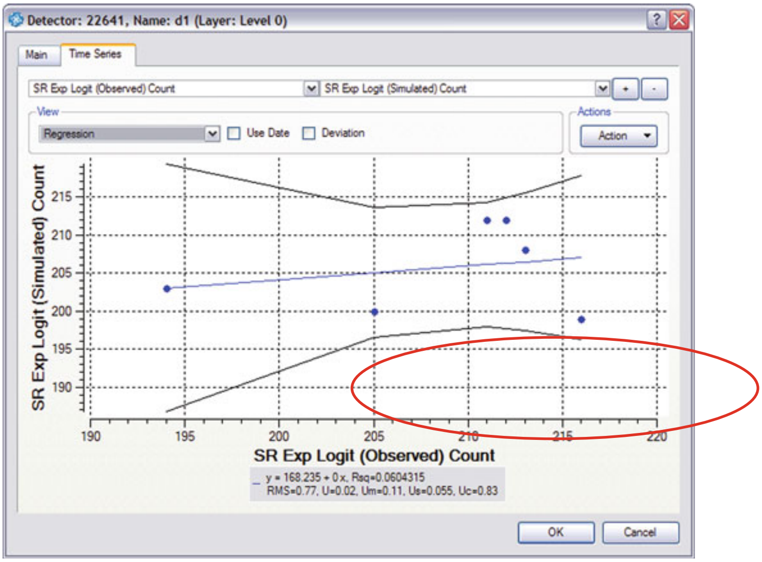


Fig. 5.12 Statistics for model validation

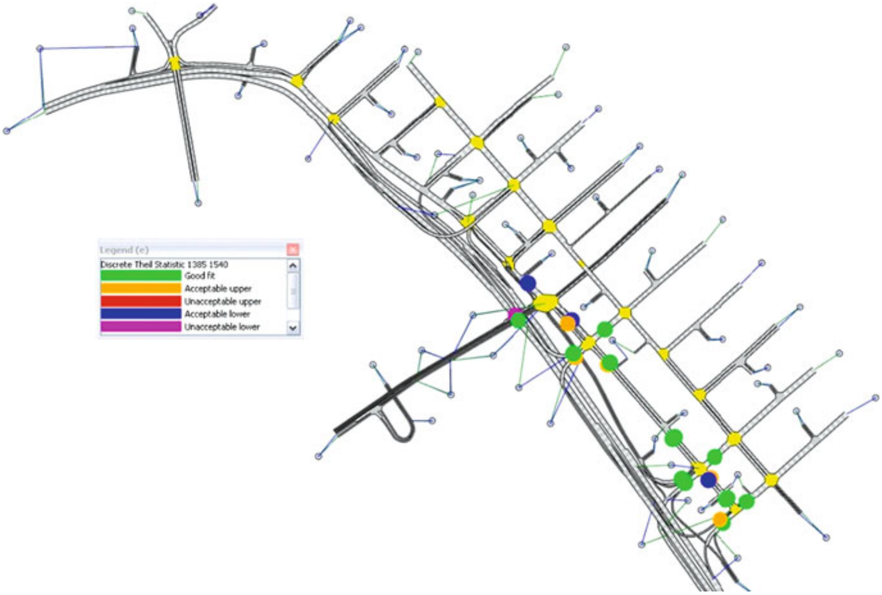


Fig. 5.13 Theil's validation global view

5.5.3.3 Comparison Based on Band Analysis

When the data collection can be automated and traffic data can be collected for long time periods (i.e. flow counts for *n* Mondays for rush hour from 7:00 until 9:00 am), the comparison between the measured data and the simulated data could consist of the comparison of two bundles of time series, the set of measured time series and the set of time series resulting from independent replications of the simulation model. Validation could then be based on developing suitable statistical procedures to compare (see Fig. 5.14) the following:

- single/mean pattern to single/mean pattern
- mean pattern to bandwidth
- bandwidth to bandwidth

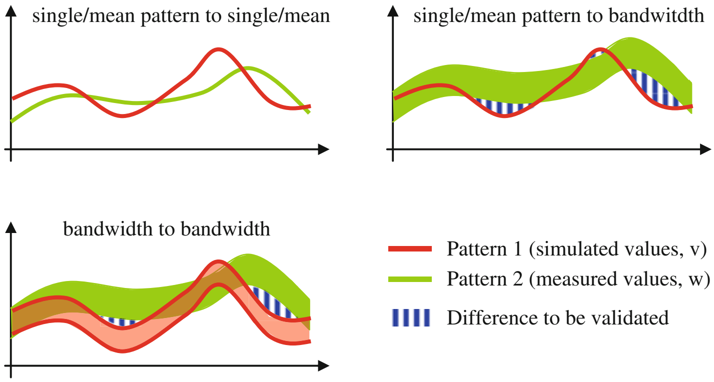


Fig. 5.14 Possibilities of comparison

5.5.4 *Calibration*

In the case of a traffic simulation model, model behaviour depends on a rich variety of model parameters. The model is composed of entities, i.e. vehicles, sections, junctions, intersections, and so on, each of them described by a set of attributes, i.e. parameters of the car following, the lane changing, gap acceptance, speed limits and speed acceptance on sections, and so on; the model behaviour is determined by the numerical values of these parameters. The calibration process has the objective of finding the values of these parameters that will produce a valid model. *Model parameters must be supplied with values. Calibration is the process of obtaining such values from field data in a particular setting and this process is completely dependent on the simulation model.* Some examples will help to illustrate this dependency between parameter values and model behaviour. Vehicle lengths have a clear influence on flows: as the vehicle lengths increase, flows decrease and queue lengths increase. In the typical car-following models, the target speed, the section speed limit and the speed acceptance, among others, define the desired speed for each vehicle on each section. The higher the target speed, the higher the desired speed for any given section, resulting in an increase in flow according to the flow–speed relationships. In this way, as part of the calibration process, one should establish for a particular model the influence of acceleration and breaking parameters on the capacity of the sections, namely for weaving sections. Similarly, depending on how the lane changing is modelled, the effects of lengths of zones where the lane-changing decision can be made influence the capacity of the weaving sections, especially when these lengths are local parameters whose values could depend on traffic condition. These effects also happen when parameters influencing the lane distribution are included in the model. Table 5.2 is an example of the influence of microscopic parameters.

The calibration process proposed in Aimsun has relationship with the type of parameter (behavioural model or dynamic traffic assignment model) and their nature (global parameters and local parameters). This process stresses the calibration of the global parameters in front of the local parameters in order to avoid the risk of entering in a overcalibration situation that deals with a situation where it is not possible to extrapolate the results obtained in a future scenario where the local calibration will

Table 5.2 Influence of micro parameters

Level	Parameter	Influence
Vehicle	Maximum desired speed Normal and maximum deceleration Maximum acceleration Speed acceptance	Speed, travel time, queue discharge, lane changing, etc.
	Minimum distance	Capacity, queue length
	Give-way time	Yield and on-ramp capacity, lane-changing blockages
	Guidance acceptance	Use of new routes
	Reaction time	Section capacity, on-ramp capacity
	Reaction time at stop	Stop and go capacity, queue measures
Global	Queue up and leaving speeds	Queue statistics
	Two lanes car following	Smoothing traffic, merging situations
	Lane-changing parameters	Distribution among lanes, interurban situations
	Speed limit	Average speed, travel times
	Turning speed	Turning capacity, travel times, average speed
Section	Visibility distance	Yield sign behaviour
	Distance zones	Turning proportions, blockings
	Distance on-ramp	On-ramp capacity, use of slow lane
	Yellow box speed	Junction capacity

be not possible because of the lack of real data. This process could be summarized as follows:

1. Calibration of behavioural models using global parameters (all vehicle type parameters, such as reaction time, reaction time at stop, speed acceptance, etc.)
2. Calibration of behavioural models using local parameters (all section and node parameters that have an influence on the vehicle behaviour, such as local reaction time variation, jam density, lane-changing zonification of the section)
3. Calibration of dynamic traffic assignment using global parameters (number of different alternatives to consider, the time interval, the default cost functions parameters, etc.)
4. Calibration of dynamic traffic assignment using local parameters (scale factor per OD pair, cost function for an individual section, etc.)

5.5.4.1 Calibration of Behavioural Models

Obviously the most exact procedure to calibrate behavioural model parameters (such as car-following model, lane-changing model and gap-acceptance model) is to conduct specific experiments in which accurate field data are recorded on the relative distances and speeds between pairs of leader–follower vehicles, and the simulation model is calibrated against the field data. A recent example of these types of experiments can be found in Manstetten et al. (1998) and Bleile et al. (1996). Unfortunately these types of experiments are expensive and can seldom be conducted in the current professional practice, but recently, inside the NGSIM project (http://www.ngsim.fhwa.dot.gov), the availability of the trajectory data in different topologies (freeways, arterial, merging, etc.) allows this type of calibration. Figure 5.15 shows the type of analysis to be conducted with respect to the speed and gap between the leader and the follower.

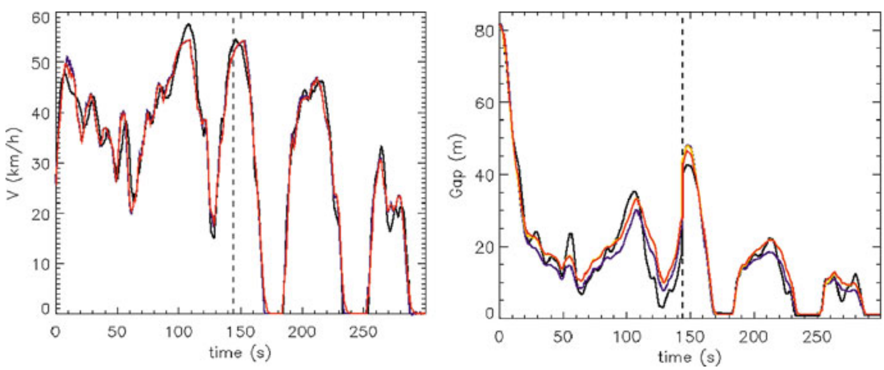


Fig. 5.15 Speed and gap profiles. Simulated versus observed

However, taking into account that correct car-following and lane-changing models acceptably calibrated must be capable of reproducing accurately enough macroscopic observable phenomena, as, for example, flow–occupancy or flow–density relationships, additional tests to analyse the quality of the microscopic simulator can be conducted to check the ability to reproduce such macroscopic behaviour. Manstetten et al. (1998) propose a test based on simulating increasing flows on a closed ring model, as the one depicted in Fig. 5.16, to reproduce a priori estimated flow–density relationships. A steadily increasing flow is injected in

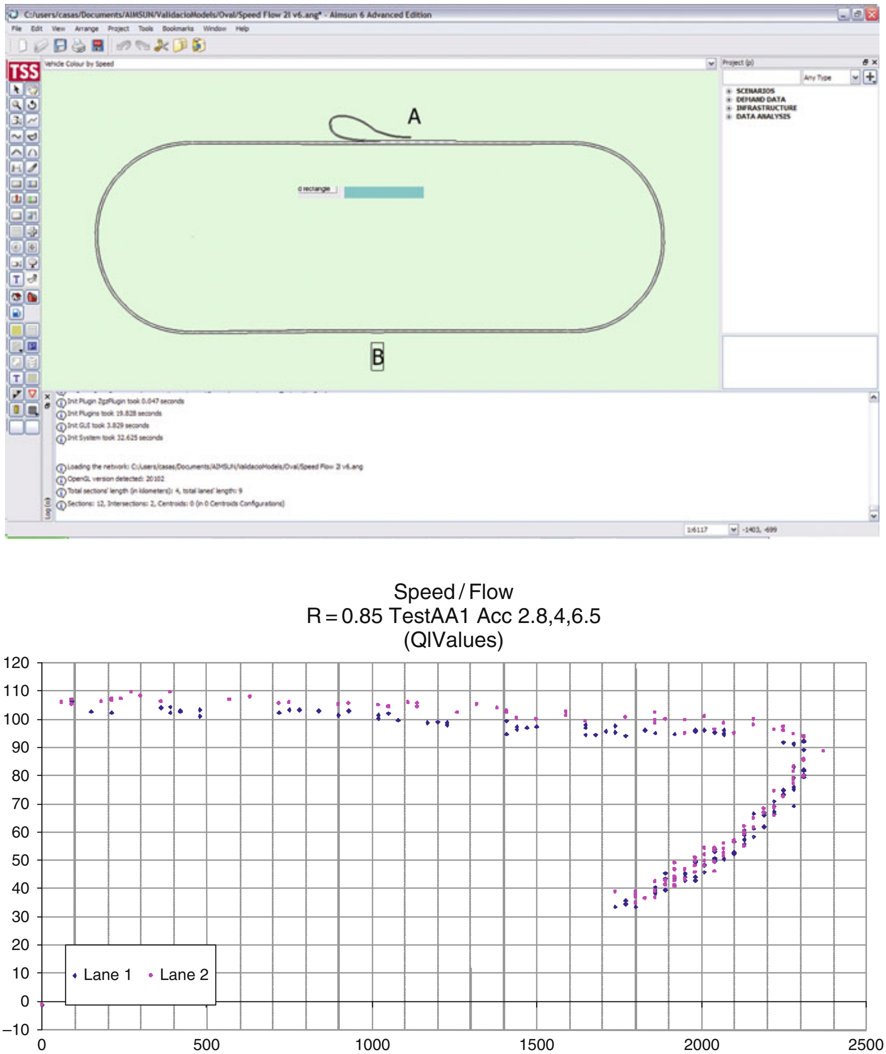


Fig. 5.16 Model to estimate speed–flow curves and example

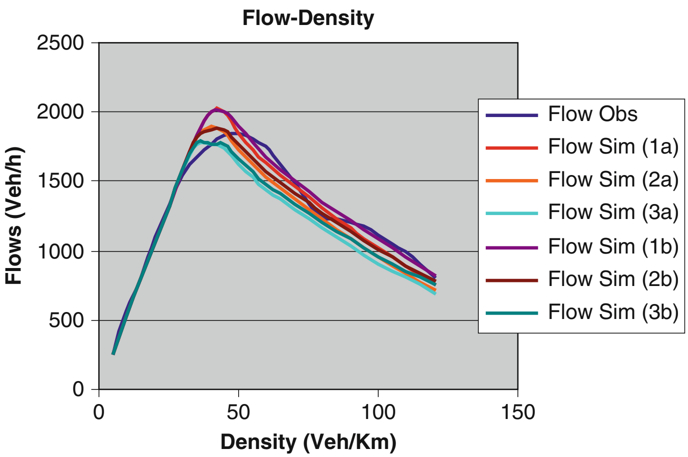


Fig. 5.17 Empirical versus simulated flow–density curves

the model at on-ramp A until reaching saturation after a predefined time horizon. A detector at B collects the traffic data. Figure 5.16 also displays the speed–flow graphics for a reaction time of RT=0.85 s, acceleration normally distributed with mean acceleration 2.8 m/s^2, standard deviation 0.56 m/s^2, normal deceleration 4.0 m/s^2 and maximum deceleration 6.0 m/s^2, providing a capacity of 2,320 vphpl.

The results of Aimsun for the simulated flow–density curve versus the empirical one for this test are displayed in Fig. 5.17, and they appear to be fairly reasonable as confirmed by the values of the RMS error measuring the fitting between the measured and the simulated values. The graphics in Fig. 5.17 also shows the sensitivity of the Aimsun car-following model to variations in the values of two model parameters, the reaction time and the minimum vehicle-to-vehicle distance (effective length). A subset of the simulation experiments to determine the values of the model parameters best fitting the observed values is summarized in Fig. 5.17, showing that the best fitting is achieved in the simulation experiment 1b with a reaction time of 0.9 s and an effective length equal to the vehicle length plus 0.75 m (Table 5.3).

Model parameters: reaction time (RT, s) and effective vehicle length (vehicle length+DM, m)

Table 5.3 Model quality as a function of reaction time and effective vehicle length

	Simulation 1a	Simulation 2a	Simulation 3a	Simulation 1b	Simulation 2b	Simulation 3b
	RT0.9/ DM1.0	RT0.95/ DM1.0	RT1.0/ DM1.0	RT0.90/ DM0.75	RT0.95/ DM0.75	RT1.0/ DM0.75
RMS	0.0645901	0.091316	0.121131	0.0518984	0.0620237	0.0920621

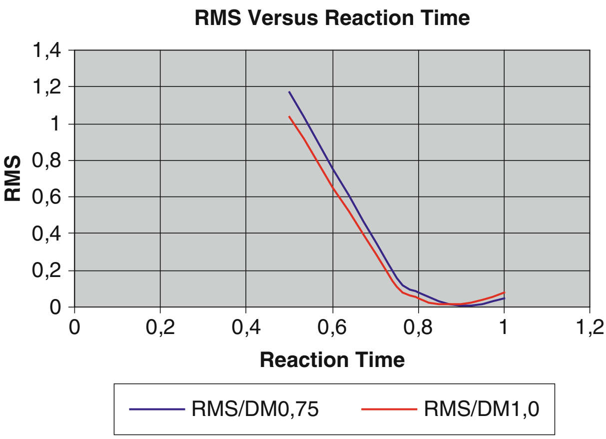


Fig. 5.18 RMS versus reaction time

The graphics in Fig. 5.18 displays the variation of the RMS error as a function of the reaction time parameter in the car-following model as implemented in AIMSUN; the blue curve corresponds to a fixed value of the minimum distance between vehicles of 0.75 m and the red one to 1 m. This illustrates in more detail the example of pilot runs of the simulation model to calibrate the parameters of the car-following model for a given context.

Similar simple models to understand and adjust the parameter values to fit the situations to study have been proposed by Yoshii (1999).

5.5.4.2 Calibration of Dynamic Traffic Assignment

Dynamic traffic assignment calibration is performed comparing traffic flows, possibly disaggregated by turning, and travel times. In urban networks the turning flows are limited by the signals (it can be roughly estimated calculating green/cycle ratio). A manual check of the reasonableness of alternative paths built between OD pairs can also be useful to detect cost errors.

An example of calibration of the dynamic traffic assignment are the computational results conducted with networks of various types and sizes (Barceló and Casas, 2006); Figure 5.19a depicts the time evolution of the Rgap function for the logit route choice function, and Fig. 5.19b depicts the plot of the Rgap versus GEH index of all replications using the logit route choice model for the network of the city of Preston in the UK which has 415 links (road sections), 165 intersections and 33 origin–destination centroids. The cloud of points that are in the area of the acceptable Rgap and GEH index represents 70% of the experiments in which the logit route choice was used. The cloud of points that are in the area of the acceptable Rgap and GEH index represents 70% of the experiments in which the logit route choice was used.

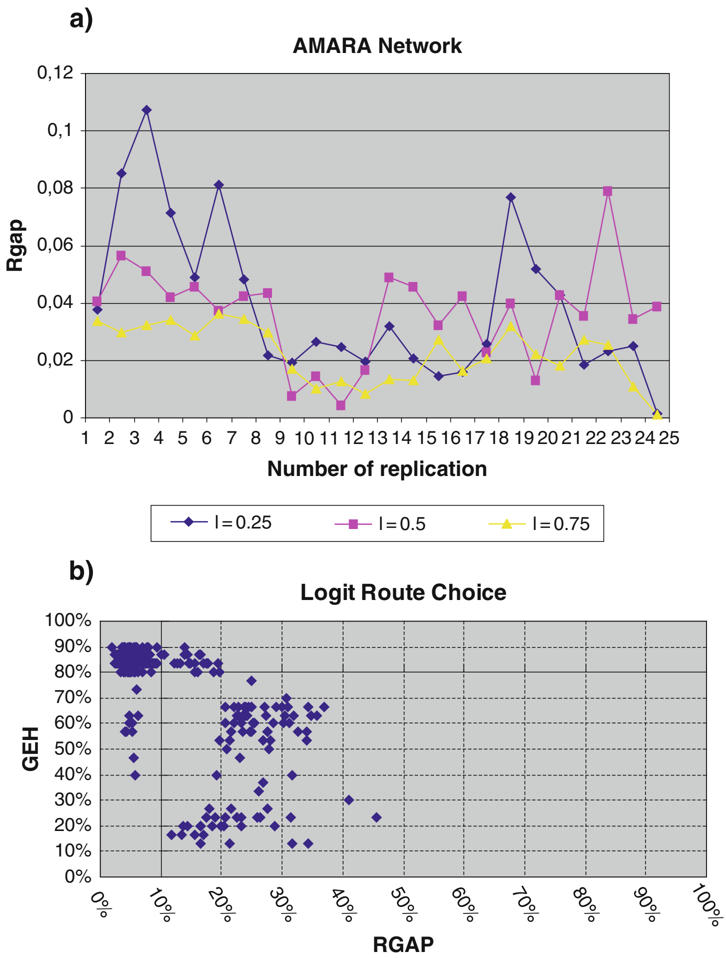
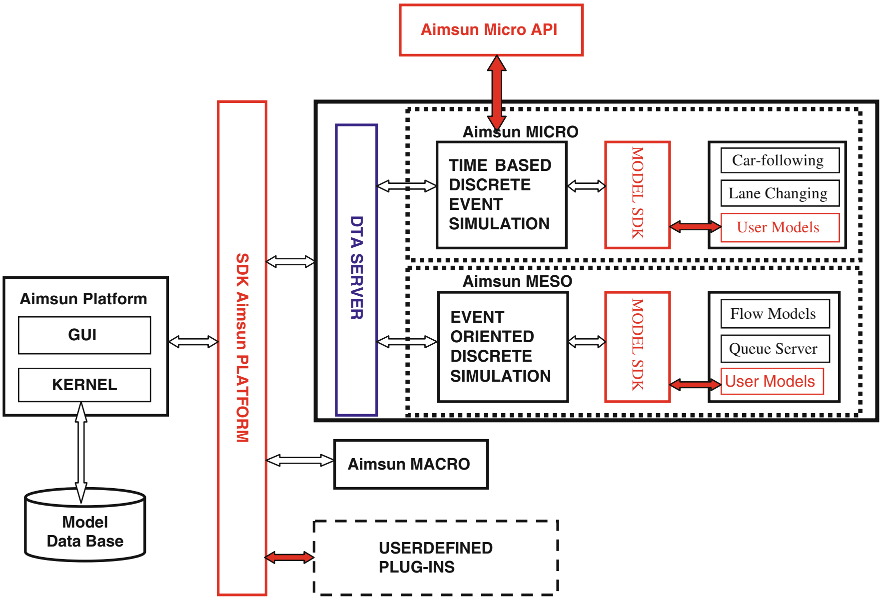


Fig. 5.19 (**a**) Rgap validation; (**b**) Rgap and GEH validation

5.6 Extended Modelling Capabilities: Working with External Applications

The functional architecture of Aimsun, shown in Fig. 5.20, allows the user different extended modelling capabilities, each one with a different role and objectives for working with external applications. The different possibilities, represented by red arrows and text boxes, are as follows:

- SDK Aimsun platform
- Micro API (APPI)
- Micro/mesomodel SDK



- SDK Aimsun Platform
- Micro API (APPI)
- Micro/Meso Model SDK

Fig. 5.20 Functional architecture of Aimsun

5.6.1 SDK Aimsun Platform

The SDK (software development kit) Aimsun Platform is a collection of libraries, header files, documents and samples that allow any user or company to develop applications for, or based on, Aimsun. This tool allows the interaction between an external application (in Fig. 5.20, user-defined plug-ins) and the Aimsun platform, which is divided into two main parts: (1) the kernel that contains all objects and their definition that are part of the application domain and (2) the graphical user interface (GUI) that contains all objects needed to implement/modify the user interface (such as dialogs, drawers and controls).

Aimsun application domain is the transportation domain. All the systems have been designed to support transportation-related applications (such as traffic simulators, location problems and assignment models). This specialization of the model offers to the developer facilities not found on other, more general, systems such as streets/roads, OD matrices, control plans and topology information. The user could develop and connect external applications allowing the interaction at level of the model definition and/or the graphical user interface. The developing language is either C++ or Python.

Examples of the use of this tool are ALMO, a software for detection analysis and traffic pattern recognition (www.momatec.de), and Legion, a pedestrian simulator (www.legion.com)

5.6.2 Micro API

The Aimsun micro API (application programme interface) is a tool or a module that gives Aimsun the ability to interface with almost any *external application* that may need access to some objects of Aimsun micro-simulator during simulation run time. The development language could be either C, C++ or Python.

This tool defines a set of functions that allow to get information from any object during the simulation time (such as vehicle information, detector measurements, statistical data, network information, demand information and traffic control plan information) and set information (such as change vehicle attributes, network object attributes, demand attributes and traffic control plan attributes). The exchange of information between the external application and the Aimsun micro-simulator is done in every simulation step.

Different applications of the micro API could be the following:

- *Traffic control/management system as external application*: The micro-simulator emulates the detection process. Then, through a set of functions, it provides the external application with the required *simulation detection data* (e.g. flow and occupancy). The external application uses this data to feed some control policy and decides which control and/or management actions have to be applied on the road network. Finally, the external application sends the corresponding actuations (e.g. change the traffic signal state, the phase duration and display a message in a VMS) to the simulation model, which then emulates their operation through the corresponding model components such as traffic signals, VMS' and ramp metering signs. The more representative examples could be the connection with the following external applications: SCATS (www.rta.nsw.gov.au), VS-PLUS (www.vs-plus.com), SCOOT (www.siemens.co.uk), UTOPIA (www.miz.it) and TUC (www.dssl.tuc.gr).
- *Vehicle-simulated data as external application*: Another uses of the Aimsun API are to access detailed vehicle-simulated data to feed some user's model (e.g. fuel consumption and pollution emissions), to keep track of a guided vehicle throughout the network by an external vehicle guidance system and to simulate the activities of vehicles such as floating cars.
- *Vehicle driving as external application*: Another uses of the Aimsun API are to access detailed vehicle-simulated data and drive a subset of the vehicles by the external application. For example, as external application, a driving simulator drives where Aimsun creates the more realistic 3D scenario for the driving simulator, according to the traffic flow theory. One example of this use is SCANeR (www.scaner2.com).

5.6.3 Micro/Mesomodel SDK

The dynamic models in Aimsun (microscopic and mesoscopic) are based on behavioural models, such as car following and lane changing, provided by default by Aimsun. The micro/mesomodel SDK is a tool that allows the implementation of new behavioural models and then during the simulation overwrites the default behavioural models. In other words, this tool could be considered as an API specifically oriented to functionalities to implement the behavioural models, such as get the leader's vehicle, get the vehicle upstream and downstream in a target lane and set a new positions and speed.

Probably the most representative example is the use of the tool to include and evaluate in Aimsun the different behavioural models developed inside the NGSIM project scope (sponsored by the FHWA).

5.7 Selected Overview of Advanced Case Studies and Applications

In this chapter we present three different applications of Aimsun to transportation engineering problems. The first two case studies focus on the use of microsimulation. The third one demonstrates the need for a combined use of the macro and micromodel. As such, it serves to illustrate the benefit of having several models integrated in the same software application.

5.7.1 The Paris Tramway

In order to improve the public transport of Paris, the French capital authorities decided to put in place a new tramway line whose itinerary will follow the so-called Boulevards des Maréchaux urban ring road. The study discussed here focussed mainly on the eastern area of the boulevards and particularly on the segment between Porte de Vincennes and Porte de Bagnolet. This tramway line will be physically separated from the boulevards with the exception of intersections. Urban planners considered four design scenarios, namely axial, bilateral and two variants of the latter.

In order to maximize modal shift to this new public facility, trams should offer competitive travel times, which means that they should not stop on signalized intersections. Consequently, tram pre-emption systems had to be designed whilst respecting that the fixed control plans is used in the rest of the city's intersections. This pre-emption system having a notable impact on the vicinity road network, the Paris City Council commissioned a study aiming to compare the four design scenarios based on the following criteria, ranked in priority order:

- Pedestrian safety
- Tramway speed

- Capacity of the crossing streets
- Capacity of the boulevards

Analytical approaches are better suited to the study of isolated intersections and are less well suited to evaluating the impacts of the pre-emption system applied on the 15 intersections of the study area. Therefore, the Paris City Council decided to undertake a traffic simulation study using the Aimsun software, and in particular the micro-simulator.

As already mentioned, the control plan of an intersection was to remain fixed while no tram was near the intersection. A fully adaptive control plan was ruled out. Thus, when a tram call is received, the current phase has to be ended (respecting the minimum green time), an inter-phase has to be activated and finally the special tram phase must be set off. Note that this tram phase has to be activated prior to the tram reaching the braking area before the intersection. Once the tram exits the intersection (exit call) and when no other tram is at the intersection, the control plan activates an inter-phase making it possible to return to the next phase of the fixed plan.

In the first phase of the study, we simulated a single intersection considered as representative (Fig. 5.21). For each design scenario, we tested several control plan options. Following the general objectives put forward by the Paris City Council, our assessment identified, for each scenario, the best control plan that would be applied to the whole network.

In addition, we analysed separately the performance of two signalized roundabouts included in the study area. The determination of the correct phases and timings was challenging: gridlocks or situations with vehicles blocking the tramway had to be carefully studied in order to get the best timings.

The final step in the analysis consisted of micro-simulations of the whole tram corridor to assess the global performance of each design and associated optimized pre-emption scheme. The first output used was the tram speed profiles in order to test if there was any deceleration other than that induced by scheduled tram stops. This allowed detecting any problem of vehicles getting trapped on the tramway or any sub-optimalities of the pre-emption system. Capacity, or better said, queue length increases have then been measured at each point in the network in order to identify bottlenecks and to evaluate the risk of congestion propagation that could lead to an intersection blockage. For each intersection, we compared the upstream demands to the downstream throughput to get the total queue increase (in number of vehicles per hour). Finally, we analysed safety aspects in terms of numbers of potential conflicts between pedestrians, bicycles (both were included in the simulation model) and cars. In addition to that, non-quantitative aspects were highlighted, such as the probability of red light violations, thanks to the input of experienced local engineers who were able to identify situations that favour such violations.

The global evaluation, based on a multicriteria approach, finally showed that the axial scenario achieves the best performance. Therefore, this design was selected for the implementation phase of the Tramway des Maréchaux (east side). The revised

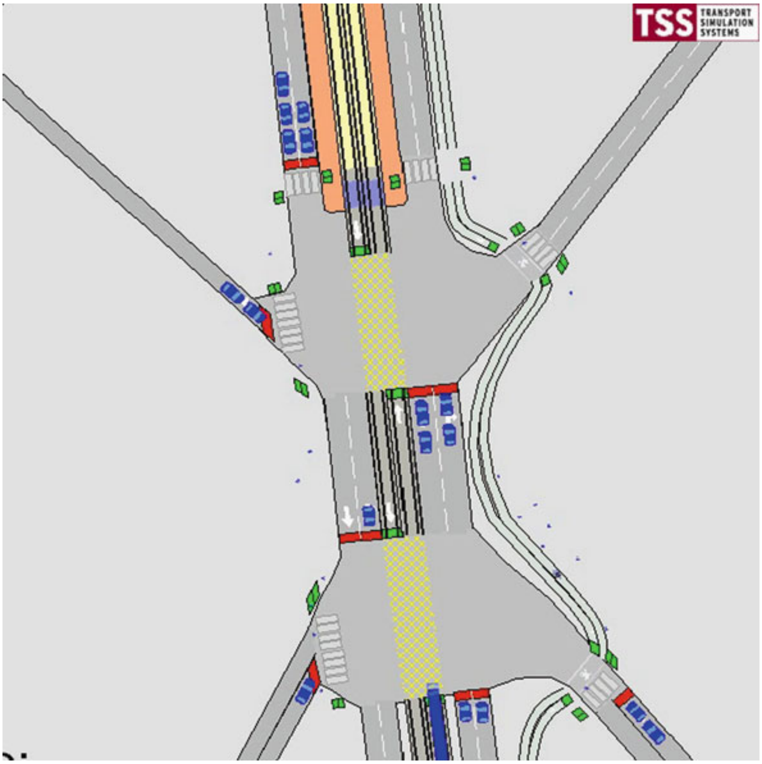


Fig. 5.21 Snapshot of a tram priority-designed intersection micro-simulation

signal control plans, including tram pre-emption, that we designed for this simulation study will be used for real implementation after some refinement related to practical limitations.

5.7.2 Behaviour-Based Highway Performance Assessment

In this section we provide an overview of a study commissioned by the Royal Automobile Club of Catalonia (RACC). The study focussed on the influence of driver lane-changing and lane usage behaviours on highway performance in the area of Catalonia.

Anecdotal evidence suggests that current driver behaviour on Catalonian highways does not follow the rules that should theoretically apply. The issue is not restricted to speed limit violations; inadequate lane changes can also be observed. The overuse of the left ('fast') lane on two-lane highways and of the central lane for three-lane highways is a frequently observed phenomenon. However, little information is available on the positive or negative implications of such behaviour on capacity, performance and safety levels. The aim of our study was to shed light on those relationships.

Forcing real drivers to behave in different ways in order to compare several types of driver behaviour in the same highway segment would be a costly, impractical and potentially unsafe undertaking. Micro-simulation in Aimsun was unsurprisingly seen as a safer, faster and much more cost-efficient approach.

To enable a quantitative assessment of highway capacity and safety levels depending on different lane-changing behaviours, we devised the following three scenarios:

- Base scenario (current situation)
- 'Legal' scenario
- 'No-rules' scenario

The first scenario reflects the behaviour that can currently be observed on Catalonian highways. The second one aims to model a situation in which driver behaviour is in strict compliance with the highway code and can be summarized as: 'Recovering the rightmost available lane when not overtaking and using the fast lane only during overtaking'. Finally, the third scenario can be thought of as the opposite extreme of the second one. No restrictions on lane usage apply, which means that drivers can use the lane they feel better on – as well as having the option of undertaking slower cars.

Using a segment of three-lane highway as a test bed, we analysed the impact of risky overtaking on flows and in particular on its potential to create congestion. To better understand the dynamics of this phenomenon of 'spontaneous congestion creation', we recorded and analysed 3D videos of the simulation.

The calibration of the Aimsun simulation parameters, based on historical traffic data set from that same three-lane highway segment, enabled us to reproduce current driver behaviour to a very high level of fidelity. The characteristic overuse of the central lane was reproduced particularly well. The data we used for calibration included distribution curves of flows per lane and flow-versus-speed data for each lane (Fig. 5.22).

Results from the simulations of all three scenarios showed that the base scenario (with 7,700 veh/h) achieves a slightly reduced capacity compared to the other two (8,000 veh/h). However, this difference being limited, an analysis using other highway sections and data sets should be done to confirm this tendency.

From a safety point of view, our analysis focussed on speed differences between lanes. The key premise of our approach was that higher values of speed difference imply lower safety levels. The idea is that speed difference increases collision probability when changing lane. The simulation results showed clearly that the 'least safe' scenario is the base one. By contrast, the 'no-rules' scenario turned out to be the safest configuration. Specifically, the latter scenario corresponded to the most homogeneous speed distribution in the traffic flow.

To measure performance, we undertook a comparison of average speeds. Our objective was to identify which scenario offers the lowest average travel time to road

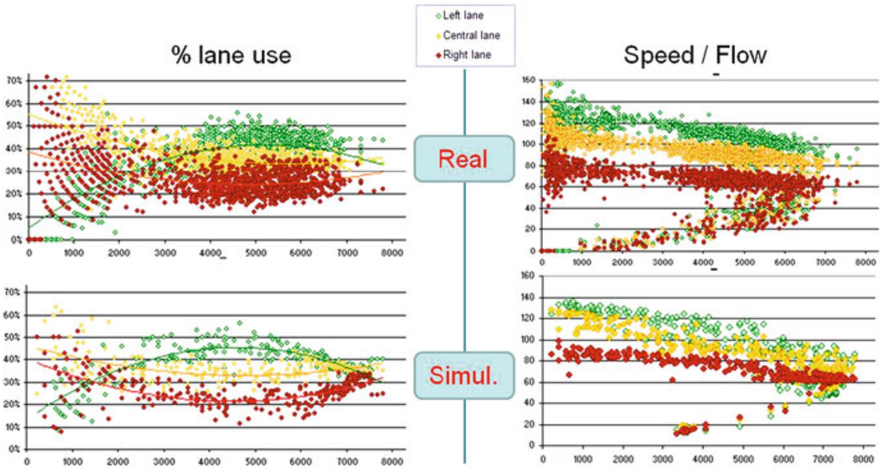


Fig. 5.22 Validation of the percentage of lane use and the speed–flow relationship for different highway traffic volumes (field data at the *top* and simulated data at the *bottom*)

users. Simulation outputs suggest that the 'legal' scenario is the best performing when this indicator is used. By contrast, the base scenario had the worst performance in this respect.

In conclusion, the current patterns of highway lane changing and lane use behaviour in Catalonia give rise to a very slight sacrifice on capacity compared to the two alternative scenarios we analysed. However, in what concerns safety and travel time, the alternative scenarios outperform substantially the existing situation. To conclude, the 'no-rules' and 'legal' approaches are better on every aspect, the main difference between them being that the former is better in terms of safety and the latter offers the best performance in travel time.

5.7.3 The Zaragoza Tramway

The use of more than one model in a traffic engineering study is becoming increasingly commonplace. A typical combination is the use of macroscopic models to determine or/and refine travel demand (origin–destination matrices) with micro (or meso)-simulation taking this data as an input for disaggregate assessments that focus on operational problems. The case study presented in this section highlights some interesting limitations in the typical 'top-down' implementation of macro/micromodelling. We discuss briefly an iterative approach which we used successfully to evaluate the future impact of a proposed tramway in the city of Zaragoza, Spain.

The root of the issue we will discuss lies in the fact that macro and dynamic models are usually not applied at the same scale level. Macromodels are typically applied to an entire city, while micro-simulation is most often used to analyse a sub-part of the same road network. To do so, OD demand for the micro area is determined using what is called a ‘traversal’ generation (reference). This consists in assigning the traffic to the general network with the static model; capturing the trips that enter, exit or stay within the subarea and finally generating the sub-matrix. The underlying risk of such an approach is that, during the microscopic simulation phase, a user could decide to evaluate changes that affect the validity of the demand matrix. Consider, for example, a new infrastructure design where an additional lane is added to a road section or the capacity of a given street is reduced by dramatically cutting the green time of a traffic signal or by implementing a bus lane. In such situations, working with the same sub-matrix for the base case and the future scenario is fundamentally incorrect. This is because if a change occurs that affects the supply conditions, it is highly probable that the traffic flows that used this part of the network will change as well. This necessitates that the demand of the subarea be computed again, moving away from the ‘top-down’ approach and adopting a macro–micro iterative one instead.

Returning to the specific case of Zaragoza in Spain, our study aim was to determine the impact of a new tramway in the city. The first step was to build a macromodel for the entire city with the purpose of determining the global OD matrix both for public transport and private cars. After calibrating and validating field measurements, we computed a traversal matrix to determine the demand in the subarea of the tramway corridor. Using the traversal demand as input, we then calibrated a micro-simulation model. The next step was to build a model of a future scenario that included the tramway line and the full-priority pre-emption system at intersections. The priority to the tram is a typical operational aspect for which micro-simulation is well suited as it allows a very accurate description of tram arrival detection and traffic signal setting changes. Once the model was ready, we micro-simulated, in the first instance using the traversal matrix of the base case. Results, illustrated in Fig. 5.23 (*left pane*), show important congestion in the centre of the area. This congestion is mostly due to the dramatic decrease of green time for the streets cutting the tramway line perpendicularly. This projection is plausible if these radical changes were to be implemented instantaneously with no notice. Taken at face value these results suggest that the new tramway will generate a critical decrease in performance for private vehicle flows. If used without further qualification, the outputs could lead to a rejection of this new public transport infrastructure initiative.

Applying the macro–micro iterative process, the street capacities that we computed in the micro-simulation of the future scenario were used as input to the macromodel and a new traversal matrix was calculated. We used the new sub-matrix to micro-simulate again the tramway corridor, providing results that can be shown in Fig. 5.23 (*right pane*). This iterative process, described in Fig. 5.24, allowed the traffic engineer in charge of this project to observe how the demand in the subarea has been adequately redistributed within the network. This re-distribution could be considered the long-term reaction of road users to the new traffic conditions. The revised micro-simulation results show now that the congestion identified in the original micro-simulation would mostly disappear. The revised results lead to a totally

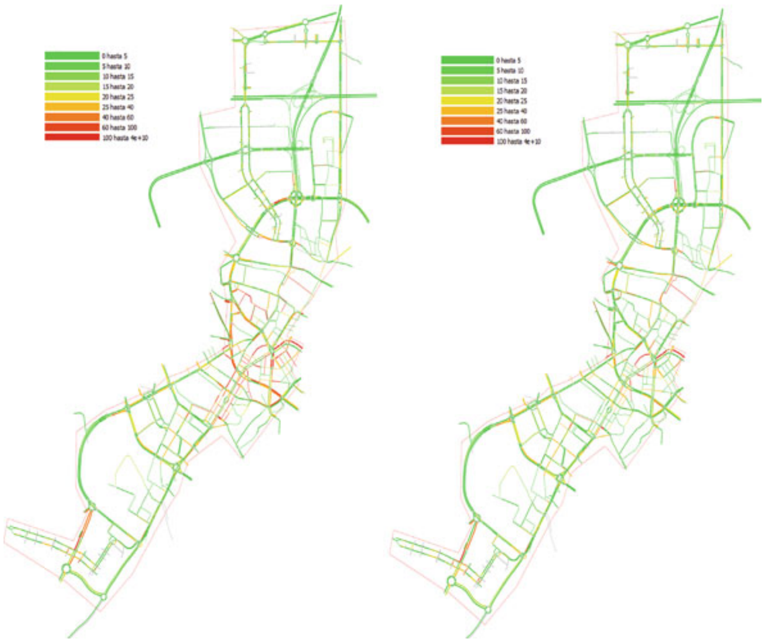


Fig. 5.23 Traffic density varying in the micro-simulated tramway corridor network before (short term, *left pane*) and after (long term, *right lane*) recalculating the local OD matrices and paths

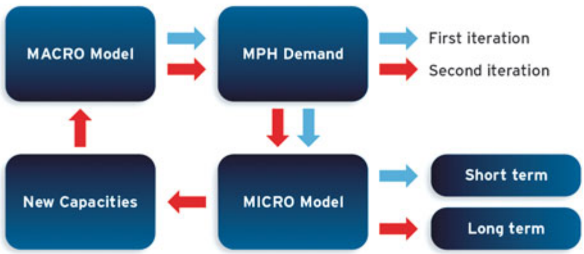


Fig. 5.24 Outline of the proposed iterative process

different conclusion whereby the new tramway has a limited long-term impact on car traffic making its implementation quite viable.

This Zaragoza case study illustrates the merit of adapting one's approach to the problem's parameters rather than following a top-down approach.

5.8 Modelling Details of Advanced Case Studies

When moving to real-time traffic management decision support, complexity increases making this an ideal field for the application of rich methodologies that apply several models sequentially, iteratively or even concurrently. To illustrate some of the possibilities, this section describes the concept of the Aimsun Online solution and discusses its implementation in Madrid as an example. In the latter part of the section, we comment on some challenges related to such applications and the development needs that they give rise to.

5.8.1 The Aimsun Online Application in Madrid

The proliferation of ITS applications makes real-time prediction capabilities a concrete requirement for the dynamic management of networks, in both urban and interurban environments. Numerous techniques for real-time forecasting have been developed and some of them have been implemented around the world. However, considering the complexity and dynamicity of problems faced by traffic managers, aggregate solutions relying on time series analysis of detector data or static equilibrium assignment techniques tend to have very limited applicability and benefit. On the other hand, traffic simulation is increasingly able to deal with very large networks, former computational and data availability limitations no longer providing a barrier to its application in real time.

In dynamic traffic management cases, the forecasting capabilities can be used to either disseminate information to travellers or, more usefully, compare the performance of different management strategies in response to a congestion and take the most appropriate action. Simulation-based systems are intrinsically better than aggregate solutions in dealing with non-recurrent events because fluctuations in supply can be explicitly factored in, and their impact under different scenarios can be quantified. These scenarios are composed of a set of actions – examples would be a lane closure, rerouting with VMS or speed limit variation – that can be activated manually or automatically based on rules which constantly process detector data. For recurring or predictable incidents, management scenarios are already implemented in the simulation model (within a scenario catalogue) and can be activated rapidly when the performance of a particular scenario has to be assessed in real time through simulation. The measures of effectiveness (MOEs) used to compare response strategies vary by project and may encapsulate safety, environmental, economic and operational considerations. A multi-objective scenario comparison, for example, may point out the scenario offering the lowest global travel time out of the ones which avoid any vehicles queuing at a specific and safety-sensitive tunnel.

A real-time simulation-based decision support tool based on Aimsun and called 'Aimsun Online' has been implemented in the Madrid traffic control centre. The newly opened M-30 urban highway (composed of a significant number of tunnel sections) is subject to many safety considerations, and many traffic evacuation and rerouting actions may be applied in response to incidents. For this reason, a tool capable of anticipating the consequences of these actions on the neighbouring network over the following critical 15 or 30 min was necessary. The tool allows operators to choose which set of actions on the city can support these safety measures efficiently while minimizing the impact on the rest of the traffic.

This application is fed field measurements in real time and uses this data to deduce the current traffic status on the streets and the actual demand (modelled as an origin–destination matrix). With signal control plans changing dynamically during the day, the application also reads the current control plan operated at each network intersection. M-30 safety actuations as well as any other incident previously detected and still existing are automatically (and in some case manually) loaded into the simulation model before starting the parallel simulation runs. Each simulation considers a concrete set of actions (strategy) that might be applied in order to improve the network situation compared to the 'do-nothing' case.

Once the parallel simulations have completed, which vary between 1 and 3 min after the initial call, operators are supplied with a summarized view of the results; this includes snapshots of predicted traffic congestion and performance indicators (MOEs). These results ultimately allow the operator to quickly see, first, if any strategies improve the situation compared to the 'do-nothing' case and if yes, which ones offer the best performance. If the suggested solution is accepted, a single validation click on the screen leads to the field application of the selected strategy. An offline-operated module allows the prediction capabilities of the simulation to be evaluated each day by comparing simulation results against real data stored during the day, offering the City Council a measure of confidence in the reliability of the system (Fig. 5.25).

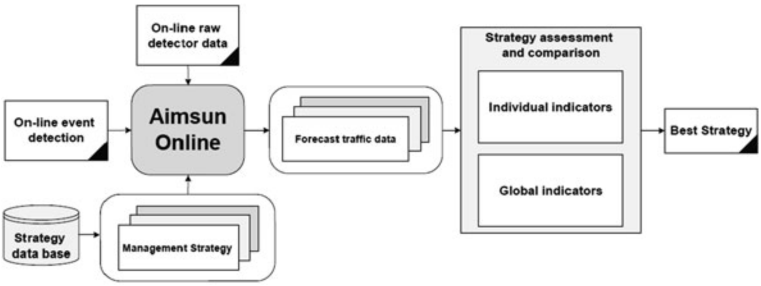


Fig. 5.25 Schematic view of the use of simulation for real-time traffic management decision support

5.8.2 Challenges and Further Needs

The Aimsun Online solution addresses a set of traffic management problems for which a combination of macro and dynamic models (micro and/or meso) is extremely well suited. The role of the macromodel here is mainly limited to OD matrix operations and more specifically, adjustments based on field measurements. However, these techniques are not without limitations and those should be borne in mind when applying them in a real-time context. To take one example, it is important to keep in mind that traffic volumes detected in the field do not represent true demand levels under high congestion; in such cases, detector data reflect only supply-side information – but not demand, which is higher. This information should therefore be treated with special attention. Using dynamic models (especially the mesoscopic model) for matrix adjustment could offer, among other advantages, an interesting solution to this problem as the demand–supply relationship would be realistically represented. Implementation of dynamic matrix adjustment solutions is therefore, in the authors' view, a clearly identified need for the future.

Another delicate aspect of such simulation-based decision support solutions is the dynamic traffic assignment. As commented in an earlier section, although the concept of dynamic user equilibrium (DUE) is well suited to describing recurring, steady-state traffic patterns, its ability to correctly represent the short-term impact on traffic distribution of a non-recurrent incident is debatable. In this case, it is indeed fair to consider the flows no longer in equilibrium and that a route choice model being able to represent the behaviour of drivers under provision of partial or complete traffic information would be more appropriate. Based on this reasoning, a careful combination of DUE-based paths and the stochastic route choice in the same run of a dynamic simulation is an ideal solution with vehicles moving from one type of assignment to the other as a function of time and information available to them.

Finally, the use of Aimsun Online always gives rise to the same debate: Which dynamic modelling approach should be used? Micro or meso? It is hardly controversial to state that each model has its own advantages and disadvantages. The microscopic simulator is able to represent section and node dynamics in detail making it suitable for ITS applications in which such granularity of information is not just useful but, one would argue, essential. In addition, a microscopic model offers a larger variety of disaggregate outputs (environmental ones, for example). However, it has the limitation of important calibration effort needs and slower computational performance. The mesoscopic approach, on the other side, is the ideal tool for fast simulations which are definitively needed in real-time applications. However, adaptive signal control and bus priority systems together with bus stops and pedestrian crossings are examples of aspects that are only approximately, if at all, modelled mesoscopically. Therefore, there is no definitive answer to this question. Depending on the network characteristics, the objectives and the level of expected accuracy micro or meso should be chosen as a compromise. However, the simultaneous use of both micro- and mesomodels coupled with appropriate combinations of dynamic traffic assignment schemes would enjoy the benefits of both categories without their limitations. In that sense, a hybrid meso–micro represents a 'best of both worlds' and as such constitutes a major need for future developments in this field and a key tenet of Aimsun's development path.

Acknowledgment The authors and the rest of the staff at TSS – Transport Simulation Systems – would like to express their sincere gratitude to Professor Jaume Barceló at the Technical University of Catalonia (UPC) for his numerous and varied contributions to the inception, design and evolution of Aimsun over the years. This chapter would exist without him.

References

Aimsun Version 6 User's Manual (2008) TSS—transport simulation systems, Barcelona, Spain, www.aimsun.com

Alexiadis V (2007) Role of simulation in corridor management. Presented at TRB2007 simulation workshop, Washington, DC, January

Balci O (1998) Verification, validation and testing. In: Banks J (ed) Handbook of simulation: principles, methodology, advances, applications and practice. John Wiley & Sons-Interscience, New York, US

Barceló J, Casas J (2002) Dynamic network simulation with AIMSUN. Presented at the international symposium on transport simulation, Yokohama (also in: Kitamura R, Kuwahara M (eds) Simulation approaches in transportation analysis: recent advances and challenges. Kluwer, 2005)

Barceló J, Casas J (2004a) Heuristic dynamic assignment based on AIMSUN microscopic traffic simulator. Proceedings of TRISTAN V, Guadeloupe

Barceló J, Casas J (2004b) Methodological notes on the calibration and validation of microscopic traffic simulation models. Transportation Research Board, 83rd annual meeting, Washington, DC, 2004

Barceló J, Casas J (2006) Stochastic heuristic dynamic assignment based on Aimsun microscopic traffic simulator. Transport Res Rec 1964:70–79

Barceló, J, Ferrer JL, Grau R (1994) AIMSUN2 and the GETRAM simulation environment. Research report, Departamento de Estadística e Investigación Operativa. Facultad de Informática, Universidad Politécnica de Cataluña

Barceló J, Casas J, Ferrer JL, García D (1998a) Modeling advanced transport telematic applications with microscopic simulators: the case of AIMSUN2, simulation technology, science and art. In: Bargiela A, Kerckhoffs E (eds.) Proceedings of the 10th European simulation symposium. Society for Modeling and Computer Simulation International, Vista, California, US, pp 362–367

Barceló J, Ferrer JL, García D, Florian M, Le Saux E (1998b), Parallelization of microscopic traffic simulation for ATT systems analysis. In: Marcotte P. Nguyen S (eds) Equilibrium and advanced transportation modeling. Kluver Acadmic Publishing, Boston/Dordrecht/London

Barceló J, Casas J, García D, Perarnau J (2006) A hybrid simulation framework for advanced transportation analysis. International symposium on traffic simulation, ISTS 2006, Lausanne, Switzerland

Ben-Akiva, M, Bierlaire M (1999) Discrete choice methods and their applications to short term travel decisions. In: Hall RW (ed) Handbook of Tranportation science. Kluver Academic Publishers, Boston/Dordrecht/London

Bleile T, Krautter W, Manstetten D, Schwab T (1996) Traffic simulation at Robert Bosch GmbH. Proceedings of the Euromotor seminar telematic / vehicle and environment, Aachen, Germany, Nov. 11–12

Carey M, Ge YE (2007) Comparison of methods for path flow reassignment for dynamic user equilibrium, May 2007. School of Management & Economics, Queen's University, Belfast, Northern Ireland

Cascetta, E, Nuzzolo A, Russo F, Vitetta A (1996) A modified logit route choice model overcoming path overlapping problems. In: Proceedings of the 13th international symposium on transportation and traffic flow theory. Pergamon Press, Oxford, UK

Ferrer JL, Barceló J (1993) AIMSUN2: advanced interactive microscopic simulator for urban and non-urban networks. Research report. Departamento de Estadística e Investigación Operativa, Facultad de Informática, Universidad Politécnica de Cataluña

Florian M, Hearn D (1995) Network equilibrium models and algorithms. In: Ball MO et al. (eds) Handbooks in OR and MS, Chapter 6, vol.8. Elsevier, Amsterdam, Netherlands

Florian M, Mahut M, Tremblay N (2001) A hybrid optimization–mesoscopic simulation dynamic traffic assignment model. Proceedings of the 2001 IEEE intelligent transport systems conference, Oakland, pp 118–123

Florian M, Mahut M, Tremblay N (2002) Application of a simulation-based dynamic traffic assignment model. Presented at the international symposium on transport simulation, Yokohama (also in: Kitamura R, Kuwahara M (eds) Simulation approaches in transportation analysis. Kluwer, 2005, pp 1–21)

Friesz TL, Bernstein D, Smith TE, Tobin RL, Wie BW (1993) A variational inequality formulation of the dynamic network user equilibrium problem. Oper Res 41(1):179–191

Gipps PG (1981) A behavioural car-following model for computer simulation. Transport Res Board 15-B:105–111

Gipps PG (1986a) A model for the structure of lane-changing decisions. Transport Res Board 20-B(5):403–414

Gipps PG (1986b) MULTSIM: a model for simulating vehicular traffic on multi-lane arterial roads. Math Comput Simul 28:291–295

Janson BN (1991) Dynamic assignment for urban road networks, Transport Res B 25(2/3):143–161

Kleijnen JPC (1995) Theory and Methodology: Verification and Validation of Simulation Models, European Journal of Operational Research, vol. 82, pp. 145–162. Elsevier, Amsterdam, Netherlands

Law AM, Kelton WD (1991) Simulation modeling and analysis, 2nd edn. McGraw-Hill, New York, USA

Liu HX, Ma W, Ban JX, Michardani P (2005) Dynamic equilibrium assignment with microscopic traffic simulation. 8th international IEEE conference on intelligent transport systems, Vienna, Austria

Lo HK, Szeto WY (2002) A cell-based variational inequality formulation of the dynamic user optimal assignment problem. Transport Res B 36:421–443

Mahmassani H (2001) Dynamic network traffic assignment and simulation methodology for advanced system management applications. Netw Spatial Econ 1:267–292

Mahut M (1999a) Speed-maximizing car-following models based on safe stopping rules. Transportation Research Board, 78th annual meeting, January 10–14, 1999 Washington DC, US

Mahut M (1999b) Behavioural car following models. Report CRT-99-31. Centre for Research on Transportation, University of Montreal, Montreal, Canada

Mahut M (2001) A discrete flow model for dynamic network loading. Ph.D. thesis, Department d′IRO and CRI, University of Montreal

Manstetten D, Krautter W, Schwab T (1998) Traffic simulation supporting urban control system development. Proceedings of the 4th world conference on ITS, Seoul

Peeta S, Mahmassani H (1995) System optimal and user equilibrium time-dependent traffic assignment in congested networks. Ann Oper Res 60:81–113

Ran B, Boyce D (1996) Modeling dynamic transportation networks. Springer, Berlin, Germany

Rouphail NM, Sacks J (2003) Thoughts on traffic models calibration and validation. Paper presented at the workshop on traffic modeling, Sitges, Spain

Sbayti H, Lu C, Mahmassani H (2007) Efficient implementations of the method of successive averages in simulation-based DTA models for large-scale network applications. TRB 2007 annual meeting, Chicago, Illinois, US

Sheffi Y (1985) Urban transportation networks. Equilibrium analysis with mathematical programming methods. Prentice-Hall, Englewood Cliffs, NY

Smith MJ (1993) A new dynamic traffic model and the existence and calculation of dynamic user equilibria on congested capacity-constrained road networks. Transport Res Part B 27:49–63

Stogios Y, Pringle R, Nikolic G (2008) A traffic simulation framework for the greater Toronto area freeway network: concept and challenges. TRB 2008 annual meeting, Washington DC, US

Theil H (1966) Applied economic forecasting. North-Holland, Amsterdam, Netherlands

Tong CO, Wong SC (2000) A predictive dynamic traffic assignment model in congested capacity-constrained road networks. Transport Res B 34:625–644

Torda A, Casas J, Garcia D, Perarnau J (2009) Traffic Simulation as a Central Element for Traffic Management Decision Support Systems. Second International Symposium on Freeway and Tollway Operations (ISFO), Hawai, USA

Varia HR, Dhingra SL (2004) Dynamic user equilibrium traffic assignment on congested multidestination network. J Transport Eng 130(2):211–221

Yoshii T (1999) Standard verification process for traffic simulation model – verification manual. Kochi University of Technology, Kochi, Japan