Final Report for Publication

SMARTEST

Contract No: RO-97-SC.1059

Project
Coordinator: ITS, University of Leeds (GB)

Partners: Universitat Politècnica de Catalunya (ES)
Mizar Automazione (IT)
CERT-ONERA (FR)
SODIT (FR)
Transek (SE)
Softeco Sismat (IT)
Högskolan Dalarna (SE)

Project Duration:
(01/03/97 to 31/05/99)

Date: 13 January 2000

PROJECT PART FUNDED BY THE EUROPEAN COMMISSION UNDER THE TRANSPORT RTD PROGRAMME OF THE 4th FRAMEWORK PROGRAMME

Table of Contents

PARTNERSHIP

The details of the project partnership and their participation status are given in the following table.

No	Role	Organisation Name	Country Code	Type	SME
1	CO	UNIVERSITY OF LEEDS	GB	AC	No
2	CR	UNIVERSITAT POLITECNICA DE CATALUNYA	ES	FC	No
3	CR	MIZAR AUTOMAZIONE SPA	IT	FC	Yes
4	CR	CENTRE D'ETUDES ET DE RECHERCHES DE TOULOUSE	FR	FC	No
5	CR	SODIT	FR	FC	Yes
6	CR	TRANSEK AB	SE	FC	Yes
7	CR	SOFTECO SISMAT S.R.L.	IT	FC	Yes
8	CR	HÖGSKOLAN DALARNA	SE	AC	No

Key:

Role: CO: Co-ordinator; CR: Contractor

Country Code: GB: Great Britain, ES: Spain, IT: Italy, FR: France, SE: Sweden

Type: AC: Additional Cost, FC: Full Cost

EXECUTIVE SUMMARY

This document is the Final Report for Publication of the SMARTEST project. This project directly addressed task 7.3/17 in the second call for proposals of the 4th Framework Programme in the Transport RTD, Road Transport Traffic, Transport and Information Management area:

Development of modelling and simulation tools to deal directly with road capacity and specific traffic management problems, such as congestion, shock-waves caused by traffic disruption, harmful emissions etc.

The project was directed towards modelling and simulation of dynamic traffic management problems caused by incidents, heavy traffic, accidents, road works, and events. It covered incident management, intersection control, motorway flow control, dynamic route guidance and regional traffic information.

This report describes the new developments and work performed during the entire project.

OBJECTIVES OF THE PROJECT

The project's objectives were to:

i) review existing micro-simulation models, so that gaps can be identified. It would build on the APAS report and other reviews such as the PROGEN report from PROMETHEUS. A State-of-the-Art review report would be produced.

ii) investigate how the existing models could best be enhanced to fill the identified gaps, thus advancing the State-of-the-Art. Prime objectives of these enhancements would be to ensure that they were transferable across Europe and that they were based on sound statistical analysis.

iii) incorporate the findings of the study into a best practice manual for the use of micro-simulation in modelling road transport and to disseminate these findings widely throughout Europe.

The main outputs of the project would be an enhanced set of micro-simulation tools for helping network managers solve their short term traffic management problems and a best practice manual detailing guidelines and procedures for their selection and use.

MEANS USED TO ACHIEVE THE OBJECTIVES

A review of existing models and simulation tools was performed to find problem areas that needed to be modelled when developing solutions to short-term traffic management problems.

Generic models and procedures were then developed and specified to fill the most important gaps. The new models and procedures were developed using existing data sets with data gaps being filled using data collected from sites in Barcelona, Toulouse, Stockholm, Turin and Leeds. Existing micro-simulation models were then enhanced to include the new models. Comparisons were made between the new model outputs and the data collected. A dissemination workshop and World Wide Web pages were produced to allow easy access to the results.

SCIENTIFIC AND TECHNICAL DESCRIPTION OF THE PROJECT

INTRODUCTION

The main focus of this project was to find out how to improve micro-simulation packages to help solve short term traffic management problems caused by accidents, heavy traffic levels, incidents, road works and events. These problems can be alleviated by using appropriate intersection and ramp metering control or by incident management or by using regional traffic information systems and dynamic route guidance. Choosing the best solutions requires the use of tools that produce realistic results in all traffic conditions. These tools need to be well validated and procedures for calibrating them and applying them consistently need to be defined. The SMARTEST project aimed to produce such tools.

A review of existing models and simulation tools was performed to find problem areas that needed to be modelled when developing solutions to short-term traffic management problems.

Generic models and procedures were then developed and specified to fill the most important gaps. The new models and procedures were developed using existing data sets with data gaps being filled using data collected from sites in:

- Stockholm
- Toulouse
- Barcelona
- Leeds
- Turin

Existing micro-simulation models were then enhanced to include the new models and data collected to assess their performance.

The SMARTEST project collaborated with the HIPERTRANS project in tackling the problems of Task 7.3/17. Links with the HIPERTRANS project and exchange of data and information was maintained by participation of appropriate SMARTEST partners at joint meetings with members of the HIPERTRANS consortium.

REVIEW OF TOOLS

A review of existing micro-simulation models that deal with traffic management problems on road networks was carried out within the project. A bibliographic search was carried out and a questionnaire was also sent to all known micro-simulation model developers. The user requirements for micro-simulation models of traffic were investigated. Data was again collected from a questionnaire, this time one was sent out to known users of road traffic micro-simulation models. Gaps that existed between current micro-simulation model capabilities and users’ requirements were identified.

Micro-Simulation Models

A bibliographic search revealed the existence of fifty-seven micro-simulation models. The models can be categorised by the traffic situation they are used to model, namely Urban, Freeway, Combined Urban & Freeway, Automated Highway Systems (AHS) or Other. These are shown in Table 1, along with the number of publications found for each model. A written questionnaire was sent out to each of the developers of these models. Thirty-two replies were received, allowing the simulation models to be analysed in a systematic fashion. Virtually all the major model developers replied to the questionnaire. The only exceptions were INTRAS, which has been superseded by FRESIM and CORSIM, the freeway models FOSIM and SIMCO2, and the AHS model SMARTPATH.

Urban	Freeway	Urban & Freeway	AHS	Other
ARTWORK (1)	***AUTOBAHN*** (4)	***AIMSUN2*** (4)	***PHAROS*** (3)	***ANATOLL*** (1)
CASIM (1)	CARSIM (2)	***CORSIM*** (2)	***SHIVA*** (1)	MIMIC (3)
CASIMIR (5)	FOSIM (5)	***FLEXSYT II*** (8)	***SIMDAC*** (1)	PARKSIM (1)
DRACULA (1)	***FREEVU*** (1)	***INTEGRATION***	SMART-AHS (1)	**TRARR** (0)
HUTSIM (7)	***FRESIM*** (7)	***MELROSE*** (1)	SMARTPATH (7)	TRGMSM (1)
MICSTRAN (2)	INTRAS (15)	MICRO (1)	SPEACS (1)	VTI (1)
MISSION (1)	***MIXIC*** (2)	***MICROSIM*** (0)		
MITRAM (1)	PELOPS (1)	***MITSIM*** (2)		
MULTSIM (1)	SIMCO2 (3)	***Paramics*** (5)		
NETSIM (78)	***SISTM*** (3)	***PLANSIM-T*** (0)		
NEMIS (9)	WEAVSIM (1)	***TRANSIMS*** (4)		
PADSIM (1)		***VISSIM*** (2)		
SCATSIM (1)				
SIGSIM (1)				
SIMNET (2)				
SITRA-B^{+} (1)				
SITRAS (1)				
STEER (1)				
STEP (1)				
THOREAU (1)				
tiss-NET WIN (1)				
TRAFFICQ (4)				

Table 1: Micro-simulation models

(Numbers in brackets indicate the number of research papers found which present results using the given model. The developers of the models in **bold** replied to the SMARTEST questionnaire)

Features Common To All The Models

Nearly all the models use a time stepping approach where the vehicles are moved around the road network using a fixed time step, typically at one-second intervals. Only three models (FLEXSYT-II, SIGSIM and SIMNET) use an event based approach where the states of objects in the network are changed at discrete times in response to events on an event list. Simple car following, lane changing and gap acceptance laws are used to govern vehicle movements along road links. Both signalised and unsignalised junctions can be modelled. Queues of traffic form at junctions and can extend all the way to upstream junctions where they can block movements.

The number of vehicles using the network is defined by specifying origin-destination (O-D) data. This is the number of vehicles that travel from each possible entrance (or origin) in the network to each possible destination. O-D data is used because it identifies the trips that need to be made in the network. The usual assumption made is that a new scheme applied to the network might have an effect of routes taken. Therefore the flows down individual links will change, but the new scheme will not have a major effect on the number of trips made from each origin to each destination. Two approaches are used to determine the routes that vehicles take through the network. The traditional method has been to define the vehicle flows on entrance links to the network and to define the percentage of vehicles that turn in each direction at the junctions within the network. Vehicles are generated randomly on the entrance links at the given rate and when they arrive at a junction a random number is generated and used to determine which direction the vehicle will travel in. For example if a vehicle arrives at a junction where 60% of the vehicles turn right and 40% turn left then if a random number between 0 and 1 is generated. If the number is less than or equal to 0.4 then the vehicle will turn left, otherwise it will turn right. An alternative method, which is becoming increasingly popular, is a route-based model. Here when each vehicle is generated in the model, it will be given a destination and an initial route from its origin to its destination by specifying which links it is to travel on to get to the desired destination. This approach has the benefits of being more realistic and being better able to cope with route changes following and incident or on supply of route guidance information. Whichever method is used, the routes or turning percentages have to be determined from the input O-D data. This can be done using an assignment model. Some of micro-simulation models have an assignment model (MELROSE, NEMIS, NETSIM, SITRA-B+, SITRAS) or a simple dynamic route choice model (MITSIM, THOREAU, TRANSIMS, Paramics) built-in. Others are closely integrated with a separate assignment model (AIMSUN2 with EMME/2, DRACULA with SATURN, MICSTRAN with DYTAM, Paramics with SATURN and TRIPS) allowing common use of inputs and outputs. Most of the rest of the models do not do any assignment themselves, they assume that this will be done using an external model.

Virtually all the simulators can model both co-ordinated and adaptive traffic signal systems. Four different approaches are used for modelling traffic signal operation. In the first approach the algorithms for changing the signal settings are an integral part of the simulator. If a particular Urban Traffic Control (UTC) system is to be modelled then code that duplicates its operation has to be written and included within the micro-simulator. FLEXSYT-II and Paramics have adopted a slightly more flexible approach. Here special signal control programming languages have been developed to allow the user to specify how the traffic signals will operate. A third approach is to treat the signal control as an external module with a well-defined communications interface to the simulator. Signal control modules can then be produced independently of the micro-simulator and linked to it as required. MITSIM and VISSIM have adopted this approach. This can be taken one step further by actually linking real UTC systems up to the micro-simulator. The simulator provides the UTC system with the data it would usually get from the real world and the UTC system sends the appropriate signal settings back to the micro-simulator. Such an approach is becoming increasingly popular and has been adopted by AIMSUN2 to link up to SCOOT, NEMIS to link up with SPOT/UTOPIA or SCOOT, HUTSIM to link up with a variety of UTC systems including SPOT, SIGSIM to link up with SCOOT and VISSIM to link up with SCATS.

Most of the models have the capability of displaying an animation of the vehicles moving round the network as the simulation progresses. Very few (AIMSUN2, FLEXSYT-II, HUTSIM, MELROSE, Paramics and VISSIM) have a graphical network builder, which can reduce the amount of time required to input the network details considerably. Most of the models provide outputs that allow efficiency indicators to be measured. These usually include travel times, travel time variability, queue lengths and vehicle speeds. About half the models now include fuel consumption and pollution emission outputs allowing environmental objectives to be assessed. Very few models produce outputs to measure safety or comfort indicators.

Most of the models are flexible in the way that key parameters can be user-defined. Integration with other models and with other databases is not so easy. One in three models is approved by a local authority or national transportation body. Typical execution speeds are of between 1 and 5 five times faster than real time.

The amount of calibration and validation of the various models is varied. Calibration data is used as an input to the model. An example of calibration data is a vehicle’s characteristics such as its acceleration and deceleration rates. Validation data is not a direct input to the model. It is used to check the output of the model. An example of validation data is the number of lane changes made on a section of road in a given time.

Urban Models

General features

	CASIMIR	DRACULA	HUTSIM	MICSTRAN	NEMIS	NETSIM	PADSIM	SIGSIM	SIMNET	SITRA-B+	SITRAS	THOREAU
ITS Functions modelled												
Co-ordinated traffic signals		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Adaptive traffic signals	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Priority to public transport		✓	✓	✓	✓	✓		✓	✓	✓		
Ramp metering			✓	✓				✓	✓			✓
Freeway flow control								✓				
Incident management					✓	✓		✓	✓	✓	✓	
Zone access control				✓	✓		✓					
Variable message signs			✓		✓				✓			✓
Regional traffic information												
Static route guidance			✓	✓	✓				✓	✓	✓	✓
Dynamic route guidance				✓	✓		✓		✓	✓	✓	✓
Parking guidance				✓					✓	✓		
Public transport information								✓				
Automatic debiting & toll plazas			✓									
Congestion pricing		✓										
Adaptive cruise control			✓	✓	✓		✓	✓	✓			
Automated highway system												
Autonomous vehicles			✓					✓				
Pedestrians and cyclists			✓									
Probe vehicles			✓		✓			✓		✓		✓
Vehicle detectors	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Objects & phenomena modelled												
Weather conditions		✓		✓								✓
Searching for parking space						✓				✓		
Parked vehicles				✓		✓	✓	✓	✓	✓		✓
Elaborate engine model					✓				✓			
Commercial vehicles		✓		✓	✓	✓		✓		✓	✓	
Bicycles / motor cycles			✓					✓				
Pedestrians			✓	✓		✓						✓
Incidents		✓	✓		✓	✓		✓	✓	✓	✓	✓
Public transport vehicles		✓	✓	✓	✓	✓		✓	✓	✓		
Traffic calming measures			✓		✓				✓			✓
Queue spill back		✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Weaving		✓	✓	✓		✓		✓		✓	✓	✓
Roundabouts		✓	✓		✓	✓	✓		✓	✓		✓
Other properties												
Runs on a PC	✓	✓	✓		✓	✓	✓			✓	✓	
Run on a UNIX machine				✓	✓		✓	✓	✓	✓		✓
Graphical Network Builder	✓		✓									
Graphical Presentation of Results		✓	✓		✓	✓	✓	✓		✓	✓	✓

Table 2: Functionality of the urban micro-simulation models

The replies from the questionnaire sent to the developers have been used to identify the main features and ITS functions modelled by each of the urban micro-simulators. These are shown in Table 2. Contact names are also supplied for each model, shown in Table 3, if further details are required.

Model	Developed or Distributed by	Contact
CASIMIR	INRETS France	Name: Simon Cohen Fax: +33-1-45-47-5606 E-Mail: simon.cohen@inrets.fr
DRACULA	Institute for Transport Studies University of Leeds Leeds, LS2 9JT, UK	Name: Ronghui Liu Fax: +44 113 233 5334 E-Mail: rliu@its.leeds.ac.uk
HUTSIM	Helsinki University of Technology Lab. of Transportation Engineering P.O.Box 2100, FIN-02015 HUT, Finland	Name: Matti Kokkinen Fax: +358(9)8031344 E-Mail: Matti.Kokkinen@traficon.fi
MICSTRAN	National Research Inst. of Police Science 6 Sanban-Cho, Chiyoda-Ku, Tokyo 102 Japan	Name: Dr. Takeshi Saito Fax: +91-3-3261-9954 E-Mail: saitot@nrips.go.jp
NEMIS	MIZAR Automazione S.p.a. Via Monti 48, 10126 Torino, Italy	Name: Carlo Di Taranto Fax: +39 11 6500444 E-Mail:100126.56@compuserve.com
NETSIM	Federal Highway Administration USA	Name: Henry.Lieu E-Mail: Henry.Lieu@fhwa.dot.gov
PADSIM*	Department of Computing Nottingham Trent University Burton St, Nottingham, NG1 4BU, UK	Name: Prof. A. Bargiela Fax: +44 - 115 - 948 - 6518 E-Mail: andre@doc.ntu.ac.uk
SIGSIM*	TORG, University of Newcastle, Newcastle-upon-Tyne, NE1 7RU UK	Name: David Crosta Fax: +44 -171 391 1567 E-Mail: davec@transport.ucl.ac.uk
SIMNET	TU Berlin. FG Strassenplanung und Strassenverkherstechnik Germany	Name: M. Glatz Fax: +49-30-314-26863
SITRA-B+	ONERA-CERT 2 avenue Edouard Belin, BP 4025 31055 Toulouse Cedex - France	Name: Jean-François Gabard Fax: +33-562-25-25-64 E-Mail: gabard@cert.fr
SITRAS	University of New South Wales Dept. of Transportation Engineering Sydney NSW 2052 Australia	Name: Peter Hidas Fax: +61-2-9385-6139 E-Mail: P.Hidas@unsw.edu.au
THOREAU	Mitretek Systems 600 Maryland Ave. SE, Suite 755 Washington, DC 20024	Name: Richard A. Glassco E-Mail: rglassco@mitretek.org

Table 3:Contacts for the urban micro-simulation models

The commonest use of micro-simulation models is for the analysis and development of urban traffic management schemes. Most of the urban models are capable of modelling the first generation of ITS functions that are now beginning to be implemented, namely responsive traffic control, static and dynamic route guidance and incident management systems.

Innovations

Some of the models cover features not included in the other models. The DRACULA micro-simulator is concerned with day-to-day variability in the network and allows sets of runs to be performed that include variability, with drivers learning from their experiences and letting it influence their route choice on subsequent runs. The models often ignore pedestrians and cyclists. Notable exceptions include HUTSIM which models both pedestrians and cyclists and MICSTRAN that has a well-validated model for the delay caused to turning traffic by pedestrians crossing the road. The SITRA-B+ model is particularly well adapted to modelling route guidance systems. It can even be connected to real external guidance systems. Only three of the models deal with roadside parking in a realistic fashion (NEMIS, NETSIM and MICSTRAN). MICSTRAN is also the only model that claims to be able to model railroad crossings within the network.

Calibration and Validation

The amount of calibration and validation of the models varies considerably. Some model developers admit to having performed little or no calibration or validation against real world data (SITRA-B+, SITRAS, THOREAU). Others have only been validated against outputs from other models (DRACULA against SATURN and PADSIM against SCOOT). Most have made some attempt to validate their models against some easily obtained datasets such as travel times. HUTSIM and NEMIS both appear to have been extensively calibrated and validated, with acceleration and deceleration and gap acceptance data having been collected for a number of different vehicle types and the models being validated against travel times, delays, stops queue lengths, speed distributions and saturation flows. NETSIM has also been well been calibrated but the data used comes from field trials in the mid 70’s. MICSTRAN has been validated against the number of lane changes made on a road section.

Limitations

The major limitations quoted by most of the urban simulation tool developers are that their models require better:

- validation;
- visualisation tools;
- links to Geographical Information Systems (GIS) for the analysis and interpretation of results from large networks;
- user friendly graphical network building tools;
- algorithms for dynamic route choice;
- modelling for public transport operations, particularly trams and priority signals.

Lack of a good model for describing behaviour at roundabouts and better input data for pollution models are also mentioned by some developers.

Freeway Models

General features

The replies from the questionnaire sent to the developers have been used to identify the main features and ITS functions modelled by each of the freeway micro-simulators. These are shown in Table 5. Contact names are also supplied for each model, shown in Table 4, if further details are required.

Tool	Developed or Distributed by	Contact
Autobahn	Benz Consult GmbH Kaiserstrasse 23 76131 Karlsruhe - Germany	Name: Thomas Benz E-Mail: Benz@s-direktnet.de
FREEVU	University of Waterloo, Department of Civil Engineering, Waterloo, Ontario N2L 3G1, Canada	Name: Dr. Bruce Hellinga Fax: +1-613-545-2128
FRESIM	Federal Highway Administration USA	Name: Henry.Lieu E-Mail: Henry.Lieu@fhwa.dot.gov
MIXIC	TNO INRO PO Box 6041 2600 JA Delft - The Netherlands	Name: Bart Van Arem Fax: +31 15 269 77 02 E-Mail: bar@inro.tno.nl
SISTM	Transport Research Laboratory Crowthorne, Berks, RG45 6AU UK	Name: Ewan J Hardman E-Mail: Mr.E.J.Hardman@T.trl.co.uk

Table 4: Contacts for the freeway micro-simulation models

Most of the freeway models have concentrated on modelling the complex geometrics found on today's freeways along with ramp metering, speed control and Variable Message Sign (VMS) systems used to manage the traffic on them. The FREEVU model has a rather more specific objective. It has been developed to assess the impact of trucks on freeway operations.

Innovations

AUTOBAHN allows a mix of traffic equipped with different automatic speed control systems to be modelled. FRESIM allows vehicles to react to static warning signs at the roadside. SISTM can model variable speed limits, which are being used in the UK to smooth traffic flows on freeways to reduce the effects of shockwaves.

Calibration and Validation

Most effort appears to have been put into calibrating and validating the driver behaviour models. AUTOBAHN and MIXIC have used data from driving simulators to calibrate their models. SISTM has been validated against flows and speed distributions in lanes on three and four lane sections of UK freeways.

Limitations

Only two of the developers mentioned any limitations in their models. The SISTM developers are aiming to improve its model of the network surrounding the freeway and the FRESIM developers would like to be able to model High Occupancy Vehicle (HOV) lanes and take account of varying lane widths.

	Autobahn	FREEVU	FRESIM	MIXIC	SISTM
ITS Functions modelled					
Co-ordinated traffic signals	✓				
Adaptive traffic signals	✓				
Priority to public transport					
Ramp metering	✓		✓		✓
Freeway flow control	✓		✓	✓	✓
Incident management	✓		✓		
Zone access control	✓				
Variable message signs	✓				✓
Regional traffic information	✓				
Static route guidance	✓				✓
Dynamic route guidance	✓				
Parking guidance	✓				
Public transport information					
Automatic debiting & toll plazas	✓				
Congestion pricing	✓				
Adaptive cruise control	✓			✓	
Automated highway system	✓			✓	
Autonomous vehicles	✓			✓	
Pedestrians and cyclists					
Probe vehicles	✓	✓			
Vehicle detectors	✓	✓	✓		✓
Objects & phenomena modelled					
Weather conditions	✓			✓	✓
Searching for parking space					
Parked vehicles					
Elaborate engine model	✓	✓		✓	
Commercial vehicles	✓		✓		✓
Bicycles / motor cycles					
Pedestrians					
Incidents	✓		✓		✓
Public transport vehicles			✓		
Traffic calming measures	✓			✓	
Queue spill back	✓	✓	✓		✓
Weaving	✓	✓	✓	✓	✓
Roundabouts	✓				
Other properties					
Runs on a PC	✓	✓	✓	✓	✓
Runs on a UNIX machine					
Graphical Network Builder					
Graphical Presentation of Results		✓	✓	✓	✓

Table 5: The functionality of the freeway micro-simulation models

Combined Models

General features

The replies from the questionnaire sent to the developers have been used to identify the main features and ITS functions modelled by each of the combined multi-purpose urban and freeway micro-simulators. These are shown in Table 7. Contact names are also supplied for each model, shown in Table 6, if further details are required.

Model	Developed or Distributed by	Contact
AIMSUN2	LIOS, Universitat Politècnica de Catalunya, Pau Gargallo 5, 08028 Barcelona, Spain	Name: Professor Jaime Barceló Fax: +34-3-401-5881 E-Mail: barcelo@eio.upc.es
CORSIM	Kaman Sciences Corp. PO Box 7463, Colorado Springs, CO 80933-7463, USA	Name: Gene Daigle Fax: +1-719 599-1942 E-Mail: daigle-cos1@kaman.com
FLEXSYT-II	Transport Research Centre (AVV) PO Box 1031 3000 BA Rotterdam - The Netherlands	Name: H. Taale Fax: +31-101-282-5842 E-Mail: H.Taale@avv.rws.minvenw.nl
INTEGRATION	Queen's University Kingston, Ontario Canada	Name: Michael Van Aerde Fax: +1-613-545-2128 E-Mail: vanaerde@civil.queensu.ca
MELROSE	Mitsubishi Electric Corp. 8-1-1, Tsukaguchi-Honmachi Amagasaki, Hyogo, Japan	Name: Yukio Goto Fax: +81-6-497-7725 E-Mail: goto@img.sdl.melco.co.jp
MICROSIM	University of Cologne Im Bruch 11a, 51427 Bergish Gladbach, Germany	Name: Marcus Rickert Fax: +49-221-470-5160 E-Mail: mr@zpr.uni-koeln.de
MITSIM	Massachusetts Institute of Technology 3 Cambridge Centre, NE20-208 Cambridge, MA 02142, USA	Name: Qi Yang Fax: +1-617-253-0082 E-Mail: qiyang@mit.edu
Paramics	Paramics Ltd 16 Chester Street, Edinburgh, EH3 7RA, UK	Name: Kim Littlejohn Fax: +44-131-220-4492 E-Mail: jkl@paramics.com
PLANSIM-T	ZPR (Centre of Parallel Computing) Weyertal 80, D-50931 Koeln Germany	Name: Christian Gawron E-Mail: gawron@zpr.uni-koeln.de
TRANSIMS	Los Alamos National Laboratory TSA-DO-SA, MS M997 Los Alamos NM 87545, USA	Name: Kai Nagel Fax: +1 - 505-665-7464 E-Mail: kai@lanl.gov
VISSIM	PTV system Software and Consulting GmbH, Stumpfstrasse 1, D-76131 Karlsruhe, Germany	Name: Dr. Martin Fellendorf Fax: +49-721-9651-399 E-Mail: fe@system.ptv.de

Table 6: Contacts for the combined urban & freeway micro-simulation models

It has been recognised that the urban models and freeway models share many features, so with a small amount of effort it is possible to convert either type of model into a multi-purpose model that can cope with either or both situations. In the case of CORSIM a different approach was used, the NETSIM urban simulator was combined with the FRESIM freeway simulator to produce the combined model. New control schemes are being developed which will effect traffic on both urban and freeway networks, for example an incident management system may divert traffic away from the urban freeway into the surrounding urban network. Therefore the ability to model both types of network with the same model is becoming essential.

	AIMSUN2	CORSIM	FLEXSYT-II	INTEGRATION	MELROSE	MICROSIM	MITSIM	Paramics	PLANSIM-T	TRANSIM	VISSIM
ITS Functions modelled											
Co-ordinated traffic signals	✓	✓	✓	✓	✓		✓	✓	✓		✓
Adaptive traffic signals	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
Priority to public transport		✓	✓	✓					✓		✓
Ramp metering	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
Freeway flow control		✓	✓	✓	✓		✓	✓			✓
Incident management	✓	✓	✓	✓			✓	✓			
Zone access control	✓		✓		✓			✓	✓		
Variable message signs	✓			✓			✓	✓	✓		
Regional traffic information								✓	✓		
Static route guidance	✓			✓	✓	✓	✓	✓	✓		
Dynamic route guidance	✓			✓	✓	✓	✓	✓	✓		
Parking guidance									✓		
Public transport information				✓							✓
Automatic debiting & toll plazas	✓		✓	✓	✓		✓	✓			
Congestion pricing				✓	✓			✓	✓		
Adaptive cruise control					✓						
Automated highway system					✓			✓	✓		
Autonomous vehicles					✓						
Pedestrians and cyclists			✓								✓
Probe vehicles				✓	✓		✓	✓	✓		✓
Vehicle detectors	✓		✓	✓	✓		✓	✓			✓
Objects & phenomena modelled											
Weather conditions							✓	✓			
Searching for parking space		✓						✓		✓	
Parked vehicles		✓			✓		✓				✓
Elaborate engine model							✓				✓
Commercial vehicles		✓	✓	✓	✓		✓	✓	✓	✓	✓
Bicycles / motor cycles			✓								
Pedestrians		✓	✓		✓						✓
Incidents	✓	✓	✓	✓			✓	✓			✓
Public transport vehicles	✓	✓	✓	✓				✓	✓	✓	✓
Traffic calming measures			✓	✓			✓	✓			✓
Queue spill back	✓	✓	✓	✓	✓	✓	✓	✓	✓		✓
Weaving	✓	✓	✓	✓	✓		✓	✓	✓		✓
Roundabouts	✓	✓	✓	✓			✓	✓	✓		✓
Other properties											
Runs on a PC	✓	✓	✓	✓			✓				✓
Runs on a UNIX machine	✓			✓	✓	✓	✓	✓	✓	✓	✓
Graphical Network Builder	✓		✓		✓						✓
Graphical Presentation of Results	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 7: Functionality of the combined urban & freeway micro-simulation models

Some schemes can cover large areas or a large number of options need to be evaluated in a short time so that control actions can be taken in real time. This means that sometimes very fast running models are needed. Some developers have therefore opted to use parallel hardware (MICROSIM, PLANSIM-T, TRANSIMS) whilst others have developed versions of their models that can run on either parallel or serial computers as required (AIMSUN2 and Paramics).

Innovations

Most of the models have similar capabilities. MICROSIM and TRANSIMS are parallel models using cellular automata.

Calibration and Validation

The amount of calibration and validation of the combined models is similar to that of the urban models. Only the Paramics model has been validated against both urban and freeway conditions. For urban conditions, outputs have been compared with measured saturation flows, for freeway conditions validation has been against lane usage, lane change rates, headway distributions and speed distributions. MITSIM and VISSIM have both been validated against data collected on freeways.

Limitations

Again the limitations are very similar to those of the urban models.

Model	Developed or Distributed by	Contact
AHS Models		
PHAROS	Institute for Simulation and Training 3280 Progress Dr. Orlando, FL 32826, USA	Name: Dr. Douglas A. Reece E-Mail: dreece@ist.ucf.edu
SHIVA	Robotics Institute Carnegie Mellon University Pittsburgh PA 15213, USA	Name: Dr. Rahul Sukthankar E-Mail: rahuls@ri.cmu.edu
SIMDAC	ONERA-CERT 2 avenue Edouard Belin, BP 4025 31055 Toulouse Cedex - France	Name: Jean-François Gabard Fax: +33-562-25-25-64 E-Mail: gabard@cert.fr
SMARTAHS	University of California - Berkeley Berkeley, CA 94720, 510-642-6000 USA	Name: Akash R. Deshpande Fax: +1-480-231-5600 E-Mail: akash@audi.PATH.Berkeley.EDU
Other Models		
ANATOLL	ISIS 11 avenue du Centre, 78286 Guyancourt, France	Name: Jean-Marc Morin
MIMIC	Chalmers Tekniska Hogskola Sweden	Name: Stig O Simonsson Fax: +46-31-772-3872 E-Mail: cvnss@vsect.chalmers.se
PARKSIM	Monash University Clayton, Victoria 3168 Australia	Name: William Young Fax: +61-3-565-4944
TRARR	Australian Road Research Board 500 Burntwood Highway Vermont South, Victoria 3133, Australia	Name: Robert Botterill Fax: +61-3-9887-8104 E-Mail: BOBB@arrb.org.au
TRGMSM	TRG, University of Southampton, Highfield, Southampton, SO17 1BJ UK	Name: Mark Brackstone Fax: +44-1703-593152 E-Mail: M.A.Brackstone@soton.ac.uk
Two-way model	Royal Institute of Technology, Stockholm, Sweden	Name: Gösta Gynnerstedt E-Mail: gynner@ce.kth.se

Table 8: Contacts for the AHS and other micro-simulation models

Vehicle-Highway Models

General features

The replies from the questionnaire sent to the developers have been used to identify the main features and ITS functions modelled by each of the AHS micro-simulators. These are shown in Table 9. Contact names are also supplied for each model, shown in Table 8, if further details are required.

Vehicle highway models are used to assess the performance of automatic intelligent cruise control systems and autonomous vehicles. These models have been designed for highly specific objectives such as the modelling of the tactical level of driving and the testing of intelligent vehicle algorithms in order to help people write Artificial Intelligence programs that drive vehicles in traffic (SHIVA) or to provide a detailed roadway environment for a simulated robot driving vehicle (PHAROS), or to evaluate the safety of a number of anti-collision devices (SIMDAC).

Innovations

All these models are highly innovative.

Calibration and Validation

No calibration or validation has been possible for either PHAROS or SHIVA as they are being used to develop systems that do not currently exist. SIMDAC contains calibrated data on driver reaction times and deceleration rates.

Limitations

All these models are much more detailed in terms of driver behaviour and network detail and run using smaller timesteps. However, they are of a much smaller scale, typically only a few junctions and links.

	PHAROS	SHIVA	SIMDAC
ITS Functions modelled			
Co-ordinated traffic signals	✓		
Adaptive traffic signals			
Priority to public transport			
Ramp metering			
Freeway flow control			
Incident management			
Zone access control			
Variable message signs			
Regional traffic information			
Static route guidance			
Dynamic route guidance			
Parking guidance			
Public transport information			
Automatic debiting & toll plazas			
Congestion pricing			
Adaptive cruise control		✓	✓
Automated highway system		✓	
Autonomous vehicles		✓	
Pedestrians and cyclists			
Probe vehicles			
Vehicle detectors		✓	
Objects & phenomena modelled			
Weather conditions			
Searching for parking space			
Parked vehicles			
Elaborate engine model			
Commercial vehicles			
Bicycles / motor cycles			
Pedestrians			
Incidents			✓
Public transport vehicles			
Traffic calming measures			
Queue spill back	✓		✓
Weaving	✓	✓	
Roundabouts	✓		
Other properties			
Runs on a PC			
Runs on a UNIX machine	✓	✓	✓
Graphical Network Builder			
Graphical Presentation of Results	✓	✓	✓

Table 9: Functionality of the AHS micro-simulation models

Other Models

Six models have been identified which do not fit into any of the other categories.

ANATOLL - a model developed to investigate the operation of toll plazas on French freeways.

MIMIC - a model to investigate vehicle interactions with other vehicles and the environment.

PARKSIM - developed to aid in the design of parking lots.

TRARR - a micro-simulation tool for investigating different rural road layouts.

TRGMSM - looking at interactions between the road network and light rail transit systems.

Two-way model - developed to test headway control, overtaking and speed control on roads with one lane in each direction, taking into account traffic coming in the opposite direction.

Contact names are supplied for each model, shown in Table 8, if further details are required.

Availability

Micro-simulation models are essentially research products. Table 10 identifies the models covered in this review that are commercial products and gives their approximate cost, it also shows the other models that can be obtained upon request from their developers.

Model	Commercial Cost (Euros)	Educational Cost (Euros)
AIMSUN2	9000	3000
FLEXSYT-II	3000	
FRESIM	275	275
HUTSIM	4000	1300
INTEGRATION	400	400
MIXIC*	Free	Free
NEMIS*	Free	Free
NETSIM	400	400
Paramics	Negotiable	Free
PHAROS*	Free	Free
SMARTAHS	Free	Free
THOREAU	Negotiable	Free
VISSIM	5000-20000	

Table 10: The cost of the available models

*Subject to licence agreement

Conclusions

Most of the micro-simulation models studied have been developed to quantify the benefits of Intelligent Transportation Systems, primarily Advanced Traffic Management Systems and Advanced Traveller Information Systems. The scale of application ranges from a small number of vehicles and intersections to a large number, about 200 nodes and many thousands of vehicles. Huge networks (300+ nodes and 1 million+ vehicles) can be considered by models that run on parallel architectures.

The models are usually used to estimate traffic efficiency in terms of speed and travel time, sometimes also considering congestion and queue length. They mainly concentrate on simulating traffic signal control, route guidance and traffic condition estimation. Each model uses its own control strategies and algorithms. Motorcycles, bicycles, pedestrians, public transport, weather conditions and on-street parking receive little attention.

Many of the models provide a Graphical User Interface (GUI) to visualise simulation results. It is generally animated and allows the evaluation of traffic conditions. A few models have a GUI to input the road network topology and other data.

Most models have only been partially validated and calibrated.

User Requirements

A questionnaire was sent out to known users of micro-simulation models in the field of transport planning, especially those involved in developing and evaluating Intelligent Transportation Systems.

A questionnaire was also placed on the SMARTEST project World Wide Web site (http://www.its.leeds.ac.uk/smartest). This section summarises the responses to the questionnaires.

The Users

A total of fifty-one responses were received from the User Requirements Questionnaire. These came from fourteen different countries, mainly from the US, UK, France and Sweden. Half of the sample represented research organisations, another quarter road authorities, 14% were private consultants and 9% manufacturers.

Main Areas of Application for Micro-Simulation Models

The respondents were fairly experienced with micro-simulation. Exactly half of the respondents were model developers themselves. About three-quarters had used simulation for modelling many applications. More than 80% of the users use traffic simulation for design and testing of control strategies. The second most common application was the evaluation of large-scale schemes (45%), while 20% of the users used traffic simulation for on-line traffic management or for evaluation of product performance.

Use of Micro-Simulation Models

General opinion

More than fifty percent of all respondents regarded micro-simulation models as necessary for analysing traffic conditions. A further third state that these models are merely useful. Only one single user believed the existing micro-simulation models to be unreliable, which seems very promising, indeed. One respondent answered: “It is a necessary tool if validated. It is an unreliable method if not validated. The aim of the model is essential as well as its limitations.” Other interesting comments were that “short time parking, very frequent marginal behaviour and pedestrian integration are difficult”, and that “they are not suitable for large travel time and large scale networks”.

Micro-simulation models used

In the survey sample of users, NETSIM is the most widely used micro-simulation model (10 users). Other models with a frequent usage are INTEGRATION (6), NEMIS (4), CORSIM (4), HUTSIM (3), VTI (3), TRARR (3) and AIMSUN2 (3). This agrees approximately with the distribution of model references discovered in the bibliographic search.

What the Users Want

Scale of application

The users were interested in modelling networks ranging in size from regional applications down to a single road. Intersection and corridor applications would both be modelled by 65% of the users. Using micro-simulation for urban network application attracts 50% of the users, so does using it for single roads. Regional traffic analysis is today only possible using parallel computer hardware or by using a macroscopic model such as EMME/2 or SATURN. But nevertheless, 23% of the respondents were interested in using micro-simulation for regional applications. Most users were only interested in investigating problems up to five years into the future, with most interest in short-term applications. Most micro-simulation runs cover a period of between five minutes up to twelve hours, with runs covering peak congestion periods (up to two hours) being the most popular. Only one user indicated that the model would be run for less than 5 minutes. Four respondents wished to use a time span of over 12 hours. The required execution speed of micro-simulation models is faster than real time. About half the users required an execution speed of over 5 times faster than real time, and another third would be satisfied with a speed of 1 to 5 times faster than real time.

Objects and phenomena modelled

The users were asked whether they thought the inclusion of a number of non-ITS objects and phenomena in a micro-simulation model were crucial, important, useful or not-important. High importance was placed on including incidents (82% crucial or important) and public transport stops

(71%) in micro-simulation models. Modelling roundabouts (59%) also scored highly. Interaction of vehicles with pedestrians and the specific behaviour of commercial vehicles scored 50%. Traffic calming methods (44%), parked vehicles (43%), bicycles & motorbikes (38%), weather conditions (34%), searching for parking spaces (25%) and elaborate engine models (15%) were considered to be less important.

Objectives and measures of effectiveness

The users were asked to indicate the importance to them of various possible outputs from micro-simulation models. These were grouped according to whether they are usually used to assess performance against efficiency, safety, comfort, environmental, or technical performance objectives.

Efficiency indicators. Travel time (94% crucial or important), congestion (84%), travel time variability (84%), queue lengths (83%), speed (77%) and public transport regularity (63%) all scored highly. Only modal split had a figure below 50%.

Safety indicators. These are also considered crucial or important to a large extent. Headway (59% crucial or important), interactions with pedestrians (59%) and the number of overtaking manoeuvres (56%) seem to be the most valuable indicators. The number of accidents (50%) and accident speeds (48%) have lower figures. Time-to-collision (52%) seems to be of surprisingly low interest in spite of its strong relationship to conflicts and accident rates. However, the comments received reflect scepticism or uncertainty about how such indicators may be produced by micro-simulation models.

Environmental indicators. Exhaust emissions (79%) are considered to be the most important environmental indicators. Roadside pollution (58%) and noise levels (59%) have lower scores.

Comfort and stress indicators. Stress (23%) and physical comfort (23%) are of little interest.

Technical performance indicators. Fuel consumption (72%) is considered very important to include as a measure of technical performance. Vehicle operating costs (45%) are less important.

Micro-simulation models will therefore be used mainly for evaluation of efficiency and environmental objectives. Safety and comfort objectives seem much less important. Another interpretation could be that the respondents were of the opinion that it is too difficult to use micro-simulation for safety or comfort assessment.

ITS or technological functions

The users were asked to indicate if it was crucial, important, useful or not important whether micro-simulation models should be able to evaluate a number of ITS or technological functions. Their replies can be ranked as follows: Adaptive traffic signals (91% crucial or important), Co-ordinated traffic signals (88%), Priority to public transport vehicles (83%), Vehicle detectors (81%), Ramp metering (78%), Incident management (74%), Variable message signs (74%), Dynamic route guidance (69%), Freeway flow control (63%). All other ITS functions (Congestion pricing, Public Transport and Regional information systems, Parking guidance, Zone Access Control, Tolling systems, Cruise control and Automated Highway Systems) were considered crucial or important by less than 40% of users.

According to the answers, micro-simulation seems to be especially valuable for the assessment of applications related to signals (adaptive and co-ordinated signals, public transport priority and ramp metering) or incidents and congestion (vehicle detectors, incident management, Variable Message Sign, Dynamic Route Guidance and freeway flow control). Urban traffic control seems to be the main application.

User friendliness

The users were also questioned about the need for a user-friendly interface for input and editing and a graphical and animated presentation of results. Virtually all the users indicated that these were crucial features. Only one respondent found ASCII tables sufficient.

Importance of model properties

The users were given a list of properties that a micro-simulation model should have and asked to rank the three most important ones. The results indicated that the most important property is that the micro-simulation model should have been validated, but also that key parameters can be user defined and that the model will run on a low cost non specialist hardware.

Conclusions

The sample of users that are included in the survey is not necessarily representative of the future model users. There is a clear geographical bias, and there is a clear bias towards research organisations. Therefore, the results should be interpreted more in an indicative than in a conclusive way. Bearing this in mind, the user requirements can be summarised as follows.

Users would like to be able to analyse a variety of specific applications, including on-line applications, control strategies, large scale schemes and product performance tests. The scale of applications ranges from regional applications to single road cases, and the time horizon ranges from on-line to several years. The requested time span of the simulations is between 5 minutes and 12 hours with an emphasis on the peak periods.

The most important requirements are demands for:

functionality - they should include the ability to model incidents, public transport stops, roundabouts and commercial vehicles,

outputs - which should give the user possibilities to obtain results in terms of

> *efficiency,* travel time, congestion, travel time variability, queue lengths, speed and public transport regularity,
>
> *safety,* headway, interaction with pedestrians, overtaking, number of accidents
>
> *environment,* exhaust emissions, noise level, roadside pollution levels
>
> technical performance, fuel consumption

ITS modelling ability - they should be able to model the following ITS functions: adaptive traffic signals, co-ordinated traffic signals, priority to public transport vehicles, vehicle detectors, ramp metering, variable message signs, incident management, dynamic route guidance and freeway flow control

user friendliness - graphical user interface for input, editing and for presentation of results

execution speed - execution times several times faster than real time

high quality performance, including default parameter values provided, key parameters user defined, validated with real data, guidelines for use provided, runs on a PC, easy integration with Database and Geographic Information Systems, short lead time before use, limited need for data acquisition and standard methods for use defined.

This is of course a tall order, and it is unlikely that all these requirements can be fulfilled within one single system. The questionnaire has nevertheless in many cases given clear indications concerning the relative importance of different factors, which is most helpful for future system development.

According to the questionnaire responses it is most important that the micro-simulation models are based on *validated* field studies. The emphasis on validation is a reflection of the uncertainty concerning behavioural relationships.

Gaps Identified

By comparing the users' requirements and the models' capabilities it is possible to discover the most important gaps that the model developers should concentrate their efforts on filling.

Model validation is a crucial issue. Users are not confident that the models have been sufficiently calibrated and validated. Therefore more real data must be collected for comparison with model outputs.

The users also appreciate the benefits of a user-friendly interface for network building and presentation of results. Few models have a network builder and interfaces with analysis packages and Geographical Information Systems could be improved.

Future micro-simulation models should ensure that they can model adaptive traffic signals, incidents and incident management systems, roundabouts, public transport priority at signals, variable message signs and dynamic route guidance. Environmental objectives are becoming of increasing importance, so models should be able to give outputs of fuel consumption and pollution emissions. All these new models will need to be well calibrated and validated.

MODEL UPDATE SPECIFICATION

Users' Requirements

The gaps in existing micro-simulation models, identified in the Review of Tools, were prioritised so that the most important gaps could be filled by enhancing the micro-simulation models under development by the SMARTEST partners.

The user requirements revealed by the survey concern several categories of features and capabilities of the micro-simulation models, but only those considered crucial or important by at least 50% of users are considered for prioritisation.

According to the users' requirements, particular attention has to be paid to the "Transport Telematics Functions". The ranking of these functions are shown in *Figure 1*.

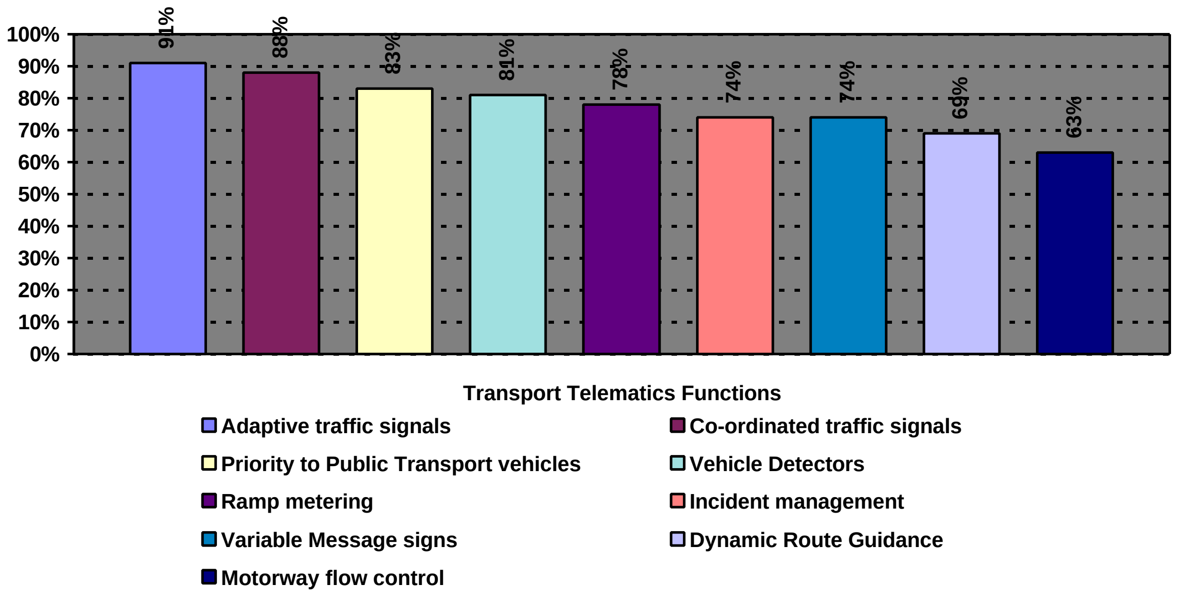


Figure 1: Users' interest in Transport Telematic Application modelling

Another important category is "traffic objects and phenomena". The ranking of functions within this category is shown in *Figure 2*. Amongst the functions in this category it clearly appears that *incidents*, *public transport* and *roundabouts* are the most important traffic phenomena to be modelled. *Commercial vehicles* and *pedestrians* are considered important by users but less important by developers.

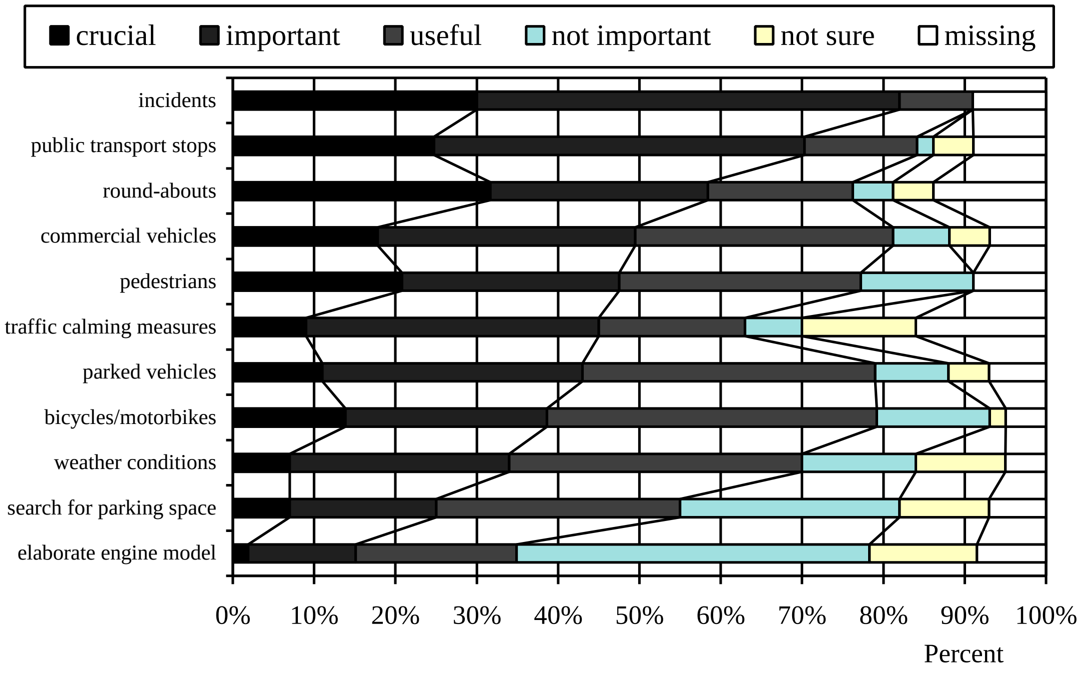


Figure 2: Users' interest in traffic objects/phenomena modelling

Users also clearly appreciate the benefit of a user-friendly interface for input and editing, and an animated presentation of the results.

An analysis of the four simulation tools being enhanced within the SMARTEST project was now carried out to determine which new features and functions could be added to the tools to help satisfy the most important users' requirements.

Indicators of efficiency and technical performance are generally already included in the SMARTEST micro-simulation tools. Safety indicators are not supported by most of the SMARTEST models even though they are considered useful by a large percentage of users.

Three of the four SMARTEST simulation tools already include an animated Graphical User Interface for presentation of results, however only one of them includes a Graphical User Interface to input the network topology and geometry data (AIMSUN2). So a great effort could be spent in this direction.

As a final comment one can state that the SMARTEST models are in a good position even though improvements are required for all of them.

Based on this analysis and taking into account the resources and of the time available in the project, the SMARTEST partners agreed to direct their effort to the items shown in Table 12

Traffic objects - phenomena

Features vs Micro Sim Models	AIMSUN2	DRACULA	NEMIS	SITRA-B+
Incidents	Yes	Yes	Yes	Yes
Public Transport	Yes	Yes	Yes	Yes
Roundabouts	Yes	Yes	Yes	No
Commercial Vehicles	No	Yes	Yes	Yes
Pedestrians	No	No	No	No

Efficiency indicators

Features vs Micro Sim Models	AIMSUN2	DRACULA	NEMIS	SITRA-B+
Travel Time	Yes	Yes	Yes	Yes
Congestion	Yes	No	Yes	Yes
Travel time variability	No	Yes	Yes	Yes
Queue length	Yes	No	Yes	Yes
Speed	Yes	No	Yes	Yes
Public Transport regularity	No	No	Yes	Yes

Safety indicators

Features vs Micro Sim Models	AIMSUN2	DRACULA	NEMIS	SITRA-B+
Headway	No	No	Yes	No
Interaction with pedestrians	No	No	No	No
Overtaking	No	No	Yes	No
Number of accidents	No	No	No	No
Accident speed severity	No	No	No	No
Time to collision	No	No	No	No

Environment Indicators

Features vs Micro Sim Models	AIMSUN2	DRACULA	NEMIS	SITRA-B+
Exhaust emissions	Yes	Yes	Yes	No
Noise level	No	No	No	No
Roadside pollution level	No	No	No	No

Technical Performance and Comfort

Features vs Micro Sim Models	AIMSUN2	DRACULA	NEMIS	SITRA-B+
Fuel consumption	Yes	Yes	Yes	No

Transport telematic functions

Features vs Micro Sim Models	AIMSUN2	DRACULA	NEMIS	SITRA-B+
Adaptive Traffic signals	Yes	Yes	Yes	Yes
Co-ordinated Traffic signals	Yes	Yes	Yes	Yes
Priority to Public Transport vehicles	No	Yes	Yes	Yes
Vehicle Detectors	Yes	Yes	Yes	Yes
Ramp Metering	Yes	No	No	No
Variable Message signs	Yes	No	Yes	No
Incident Management	Yes	No	Yes	Yes
Dynamic Route Guidance	Yes	No	Yes	Yes
Motorway Flow Control	No	No	No	No
Congestion Pricing	No	Yes	No	No

Table 11: Modelling Capabilities of the SMARTEST tools

traffic phenomena modelling	Public Transport
	Roundabout
	Traffic Calming
	Parking Management
Transport Telematic Functions	Adaptive Traffic Signals
	Public Transport Priority
	Vehicle Detectors
	Variable Message Signs
	Dynamic Route Guidance
	Incident Management
	Ramp Metering
user friendly (graphical) interface	Results Analysis
	Network Builder
Better Validation	

Table 12: The most important user requirements

Table 13 summarises improvements and new implementation planned for the SMARTEST tools in the lifetime of the project.

	AIMSUN2		DRACULA		NEMIS		SITRA-B+	
Public Transport Services			✓	☺	✓	☺	✓	☺
Roundabout			✓	☺	✓			☺
Traffic Calming				☺	✓			
Parking Management							✓	☺
Adaptive Traffic Signals	✓	☺	✓	☺	✓	☺	✓	☺
Public Transport Priority			✓	☺	✓	☺	✓	☺
Vehicle Detectors	✓		✓	☺	✓	☺	✓	
Variable Message Signs	✓	☺			✓	☺		☺
Dynamic Route Guidance		☺			✓	☺	✓	
Incident Management	✓	☺					✓	☺
Ramp Metering	✓	☺						
Network Builder				☺				
Results Analysis		☺			✓	☺		
Better Validation		☺	✓	☺		☺		☺

✓	already exists
☺	to be implemented or to be improved

Table 13: Improvements and new implementations in the SMARTEST models.

Planned enhancements to the SMARTEST Tools

In the following paragraphs the enhancements which were planned for each micro-simulation tool are described in more detail.

AIMSUN2

The following functions are to be developed or enhanced in AIMSUN2:

- *Incident Management*
- *Adaptive Traffic Signals*
- *Ramp Metering*
- *Variable Message Signs*
- *Dynamic Route Guidance*
- *Results Analysis Tool*

Improvements to the incident generation model will include deterministic and random incident generation. Deterministic incidents will be defined either through the user's interface or by means of an incidents log file. Random incidents will be generated according to certain random distributions that can be variable according to certain section characteristics.

The adaptive traffic signals improvements will consist of a new and more flexible definition of the traffic control plans and the development of a new interfacing protocol between AIMSUN2 and any external traffic control or management application. This link will be implemented by the use of Dynamic Link Libraries (DLL) through which any user will be able to either implement or communicate any control or management strategy.

Through this interfacing protocol it will be possible not only to control any traffic signal but also any ramp metering or Variable Message Sign.

Regarding VMS and Dynamic Route Guidance Systems, a better behavioural model that emulates the influence that routing information may have on the drivers will be implemented. To achieve a better characterisation of the drivers, several former global parameters will be transformed into local or individual parameters (i.e. compliance level and speed acceptance parameters).

A new Result Analysis Tool will be developed. Its main functionalities will be to define and conduct simulation experiments, to perform results analysis and make data representation and to provide statistical tools for model calibration and validation.

DRACULA

Five models will be improved in DRACULA:

- *Roundabouts*
- *PT Services*
- *Adaptive Traffic Signals*
- *PT Priority*
- *Detectors*

one new model will be added:

- *Traffic Calming*

Improved validation of car-following, lane changing and gap acceptance models will also take place. To aid user friendliness, the possibility of adding an improved Windows based interface and of using the GETRAM network builder will also be investigated.

Improvements in the PT services model will include a new bus stop model and the development of guided bus and tram operations. New roundabout and traffic calming models will also be developed, which will be calibrated and validated using data collected in Leeds.

The adaptive traffic signals improvements will concentrate on linking DRACULA to a BALANCE UTC system that is due to be installed in Leeds and Sheffield. The installed BALANCE system is planned to use the new NTCIP communications protocol to link up its various components. With this in mind a DRACULA interface that also uses NTCIP will be developed. The improvements in the detector model in DRACULA will concentrate on providing the BALANCE system with the on-street information it requires. As well as the usual loop detector data this will also include both public transport and emergency vehicle location information. PT Priority will look at the priority measures to both buses and trams that are provided by the BALANCE system. A test network in London will be used to calibrate and validate the new models.

NEMIS

Two models will be improved in NEMIS:

- *Public Transport Services*
- *Vehicle Detectors*

Results Analysis will be improved from the point view of both indicators and graphics representation.

Main efforts will be spent to improve and standardise the interface between the micro-simulation model and the external Transport Telematic Applications. This activity involves:

- *Adaptive Traffic Signals*
- *Public Transport Priority*
- *Variable Message Signs*
- *Dynamic Route Guidance*

The standard interface will be based on a TCP/IP communication protocol that will be adopted to connect the computer where the model runs to the network where the external strategies will operate.

Public Transport Services model will be tested onto a common scenario involving the UTC controlled area. Data are available from the SIS AVM system.

Vehicle detector validation concerns performance, error rate and breakdown occurrence. Data are available from the UTOPIA maintenance statistics.

The validation of the standard interface concerns mainly operational aspects. Stress conditions will be generated connecting the model to a real network of SPOT traffic control units.

Further validation activities are envisaged that concern the calibration of the car following rule according to data collected from the field.

Further parameters such as driver compliance to VMS and DRG indications will be calibrated against the information made available by surveys conducted in the test-site by other specific projects.

SITRA-B+

The following functions are to be developed or enhanced in SITRA-B+:

- *Roundabout*
- *Public Transport Services*
- *Incident Management*
- *Adaptive Traffic Signals*
- *Public Transport Priority*

- *Variable Message Signs*
- *Parking Management*

Two of these functions are to be submitted to a detailed validation plan. They are:

Roundabout: validation concerns lane changing in the roundabout, lane choice at roundabout entrance and driver behaviour entering the roundabout (gap acceptance)

Public Transport Services: validation concerns bus behaviour along the route and at bus stops (waiting time, travel time).

Other functions to be developed or enhanced in SITRA-B+ will be tested according to the verification tests described in the model update specifications. Note that:

For *Incident Management*, the incident time, place and duration will be tested to perform as specified. Driver reaction to the management actions has been already tested in SITRA-B+ as long as these actions are stop signs, traffic lights, speed limits and reserved lanes for incident response units. Validation of *Variable Message Signs* concerns the determination of user compliance rates. Driver interviews downstream the VMS are planned to be performed in 1998 on a radial axis of the Toulouse Test Site. They could be used for model calibration if data are available. Validation does not really apply for *Adaptive Traffic Signals*. It would rather concern the adaptive strategy itself. The verification tests described in the model update specifications will be performed. *Public Transport Priority*, tests related to the external strategy or to the communication process are not considered. Verification tests will be performed. *Parking Management* validation requires data such as car park occupancy rates, average travel time of vehicles in search of a parking space and average of other vehicle travel time down links containing car park entries. Such data are not available for the Toulouse Test Site.

MODEL DEVELOPMENT

Introduction

The new modelling features and improvements that were identified as gaps and then prioritised, were developed according to detailed requirements specifications.

AIMSUN2

Introduction

Most of the new models included in AIMSUN2 are based or make use of the GETRAM Extension Module, a set of Dynamic Link Libraries (DLL) through which any user is able to either implement or communicate any control or management strategy to AIMSUN2.

GETRAM Extensions

The current trend in the development of Advanced Transport Telematic Applications, either real-time adaptive, or based on other specific approaches, is far from being standardised. To try to incorporate them in a microscopic traffic simulator in a specific fixed way would therefore be of little use. If any specific ATT application were included in a micro-simulator as an in-built function, it is likely that it would not be suitable for simulating other similar applications.

This is true whenever we address the problem of simulating traffic management and control systems such as, for example:

- adaptive signal control systems (SCOOT, SCATS, SPOT/UTOPIA, PRODYN, BALANCE etc),
- vehicle actuated control,
- public transport priority systems,

- Advanced traffic management systems (using VMS, traffic calming strategies, ramp metering policies, etc),
- Vehicle guidance systems,
- Public transport scheduling and control systems,
- applications aimed at estimating and controlling the environmental impacts of pollutant emissions, and energy consumption.

The main question then is: "How can these Advanced Transport Telematic Applications be properly evaluated and tested by simulation?"

To evaluate and test any of these systems a micro-simulator must be capable of incorporating in its model the traffic devices that are used by the system: e.g. detectors, traffic signals, VMS, etc. It must also emulate their functions: e.g. provide the specific traffic measurements at the required time intervals, increase the phase timing in a given amount of time, implement a traffic calming strategy (slow down the speed on a road section, recommend an alternative route, etc). How can such evaluations be done by simulation without explicit in-built modelling of the specific Advanced Telematic Application?

The approach taken in GETRAM/AIMSUN2 consists of considering the Advanced Telematic Application to be tested as an *External Application* that can communicate with AIMSUN2. An ad hoc version of AIMSUN2 including a DLL has been developed. This library gives AIMSUN2 the ability to communicate with almost any of the above-mentioned external applications.

Using the Tedi and AIMSUN2 functions, vehicles, traffic signals, detectors, ramp-meters and VMS and can be modelled and their attributes defined. The process of information exchange between AIMSUN2 and the external application is shown in Figure 3:

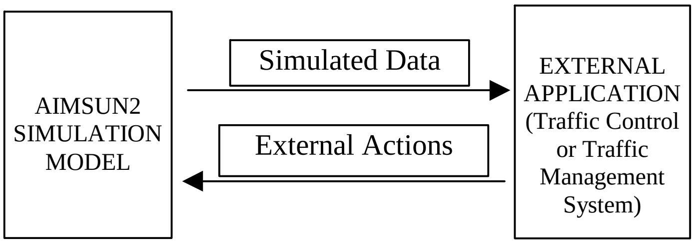


Figure 3: Process of information exchange

The AIMSUN2 model of the road network emulates the traffic providing the external application with the required "*Simulated Data*", which may be vehicle data, detector data, statistical data or control data. The external application (user provided) decides which control, management or other actions have to be applied on the road network and sends the corresponding information to the simulation model which then emulates their operation through the corresponding model components such as vehicles, traffic signals, VMS, etc.

The GETRAM Extensions are implemented using DLL's (Dynamic Link Libraries). There are two modules: on one side there is the executable program which corresponds to the simulation logic and on the other side we have a DLL (or a set of DLL's) which corresponds to the control and management logic (or policy).

The DLL has to have four functions defined:

1. GetExtInit(): It is called when AIMSUN2 starts the simulation and can be used to initialise whatever GETRAM Extension needs.
2. GetExtManage(float time, float timeSta, float timTrans, float acicle): This is called in every simulation step at the beginning of the cycle, and can be used to request detector measures, vehicle information and interact with junctions, meterings and VMS in order to implement the

control and management policy. This function receives four parameters in relation to time: absolute time of simulation, time of simulation in stationary period, duration of warm-up period, duration of each simulation step.

3. GetExtPostManage(float time, float timeSta, float timTrans, float acicle): This is called in every simulation step at the end of the cycle, and can be used to request detector measures, vehicle information and interact with junctions, meterings and VMS in order to implement the control and management policy. This function receives four parameters in relation to time: absolute time of simulation, time of simulation in stationary period, duration of warm-up period, duration of each simulation step.
4. GetExtFinish(): It is called when AIMSUN2 finish the simulation and can be used to finish whatever GETRAM Extension needs.

The next figure shows graphically how AIMSUN2 and a GETRAM Extension DLL interact.

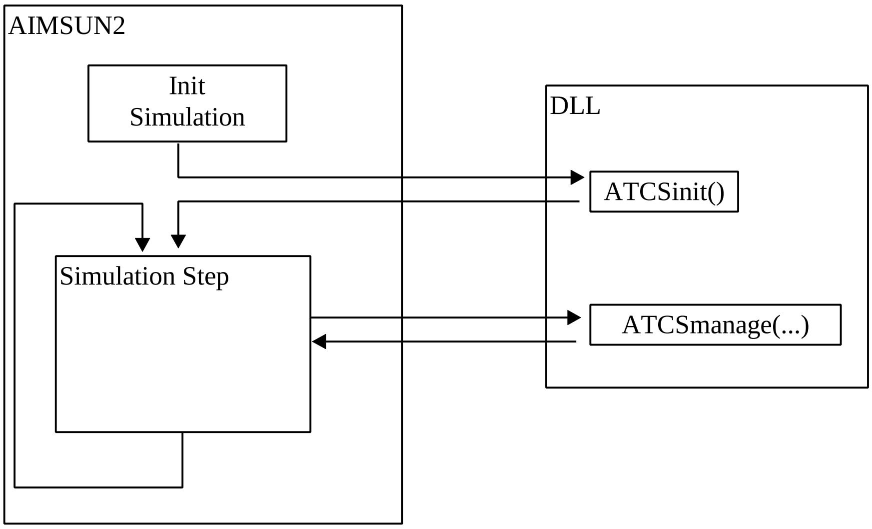


Figure 4: How AIMSUN2 and GETRAM Extensions interact

The functions provided by the simulator that can be called by the DDL to perform the interaction between AIMSUN2 and the GETRAM Extension can be grouped into different sets, depending on the type of information they are related to:

- junction control,
- ramp-metering,
- VMS,
- detectors,
- vehicle information,
- vehicle generation,
- vehicle tracking,
- statistics.

The DLL can be built from the supplied files using a C++ compiler. After building the DLL with C++ compiler, it has to be placed in the same directory where AIMSUN2 is located.

Incident management

In AIMSUN2 the microscopic traffic simulation of the road network emulates the traffic detector measurements used by an incident detection algorithm. The incident detection algorithm is then a second component of the simulation model, a component that would be supplied by each user according to their current or foreseen practice.

The proposed open platform requires an interface to integrate the two components (traffic simulation model and incident detection model) which could consist of an exchange of detector measurements according to the degree of aggregation and format required by the user.

A third component of the common simulation, according to this approach as open platform, is the traffic management/incident management component, whose integration with the two other components is illustrated in the diagram of Figure 5.

The interfacing between AIMSUN2 and the incident detection and management system is achieved through the GETRAM Extension module, described in the previous section, through which the user is able to implement and communicate any external application to AIMSUN2.

The simulation model emulates traffic flows at the network, and generates incidents according to the specified patterns. The emulation of the detector measurements defines for the detection algorithms an input equivalent to the input that the real detectors supply. This procedure enables the estimation of the detection time taking into account that the simulator knows which is the exact time at which the incident was generated.

The incident detection module communicates to the traffic management and incident response systems the occurrence of the incident and its location. The specific management and response actions, such as motorist information using variable message panels, access control using ramp metering policies, speed control on the main road sections, etc., are decided by the traffic management module and communicated to the simulation model which implements them. The subsequent simulation experiments enable the assessment and evaluation of the impact of the proposed actions.

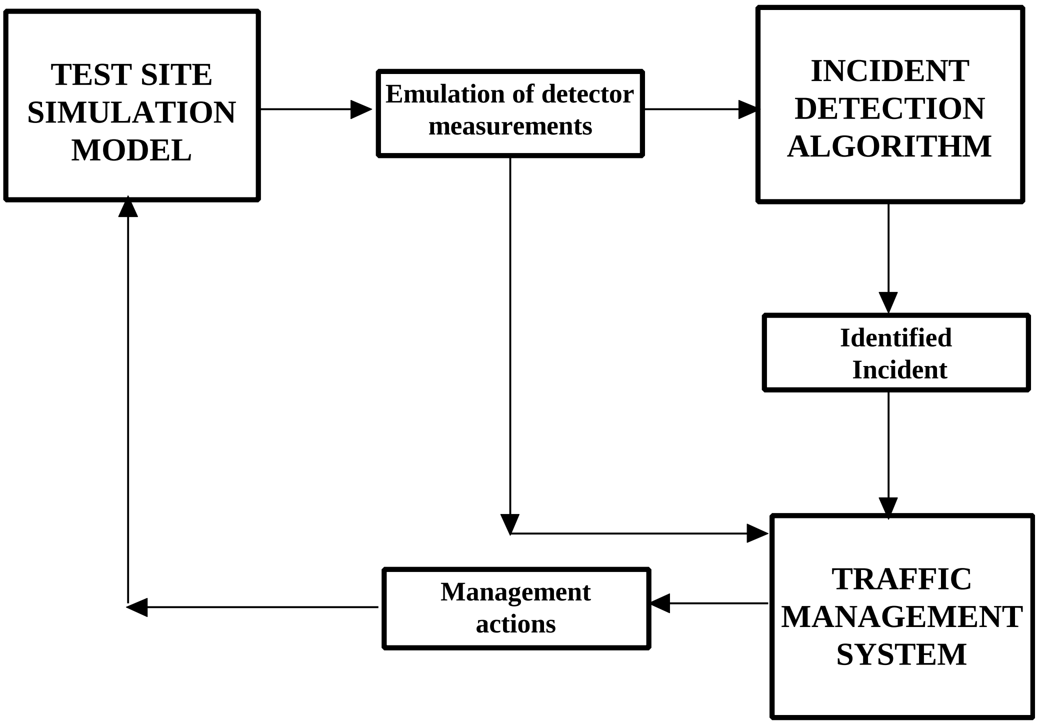


Figure 5: Data flows between the simulator and the detection module

Traffic volume, occupancy, space mean speed and density are data collected by the AIMSUN2 traffic sensors. The detection process can be either based on the direct observations or on more complete information from treated data, e.g., first and second order statistics of the data. Traffic data may include travel time and routing information, e.g., turning movements or tracking of vehicle paths through the test site. Data are sampled at regular intervals. Traffic data are statistically treated and processed.

The model is capable of generating incidents anywhere on the simulated road network and then reproduces the dynamics of the queue and congestion building processes. If an incident prediction or incident warning system is taken into account, then incidents should be created on each road section according to the corresponding probability model for that section.

The simulation process deals with the probabilistic incident generation as a scheduled event for the sections. That means that at the beginning of each simulation step, at the same time that the simulator control module updates the other scheduled events, such as those related to the traffic control signal changes, it will also check whether incidents will occur at the sections according the corresponding probability distribution. The fields that compose an Incident Event are: Time at which incident will take place, Duration of Incident, Number of lanes blocked by the incident and Length of incident.

Therefore, Traffic Incident Generation will follow an Event Scheduling simulation approach. At the beginning of simulation, the first Incident Event will be generated for every section in the network, according to the specified probability distribution.

Incidents are sorted in the simulation Event List by time of occurrence. At every simulation step, the Event List is checked to see whether or not a traffic incident is due to occur in the current simulation step. If so, the incident is generated and the corresponding event is removed from the Event List. Then, next incident event is scheduled for that section, according to the incident generation parameters defined.

The incident is implemented in the simulation by the generation of a dummy vehicle, which is stopped at the incident position for the duration of the incident. Therefore, other vehicles will be affected by the incident by following normal vehicle behaviour models (car following and lane changing). A check is carried out when updating vehicles to see whether an incident has finished. If the vehicle is an incident dummy vehicle and the incident duration time has expired, the dummy vehicle is removed from the network, thus removing the blockage.

The simulator produces detection output data periodically, provided that there are any detectors defined in the network. The data produced depends on the measuring capabilities of the detectors. It may be Count (number of vehicles per interval), Occupancy (percentage of time the detector is covered) and Speed (mean speed for vehicles crossing the detector). These data may be stored in files or directly accessed by the Incident Management System, through the GETRAM Extension Module.

The types of management actions that are modelled include modifications to speed limits, recommendations of alternative routes or just information about the presence of an incident. The modeller can use any of the following actions as a response for these messages:

- Modifications of the speed limit of any section. This is used to model both, the variable speed limit signs and the warnings for incidents or congestion ahead,
- Input flow modifications, which is only applicable to the input sections. The modeller can specify an increment or decrement (in percentage) in the flow rate.
- Re-routing actions. Depending on the type of simulation, based on turning proportions or in O/D matrices, they are turning proportions modifications for any section in the network, modification of next turning movement for drivers in certain section going to specific destinations, or modifications of destination centroids.

Adaptive signal control

The approach used in AIMSUN2 to model Adaptive Traffic Signals is by means of the GETRAM Extensions Module. In this way, AIMSUN2 traffic signals are adaptive if there is an Adaptive External Traffic Control System interfaced to AIMSUN2 that is running during the simulation and it takes control of the signals.

For the intersection control, AIMSUN2 uses a phase-based approach in which the cycle of the junction is divided into phases where each one has a particular set of signal groups with right of way at the same time.

All the turning movements that are controlled by the same traffic signal and have right of way simultaneously can be grouped in one signal group. Then, a sequence of phases is defined for the whole junction. Each phase has a set of signal groups associated with it.

During the simulation of a scenario, AIMSUN2 executes a fixed control plan taking into account the phase modelling for each junction. However, this fixed control definition can be variable within the simulation period. The user may specify different fixed plans that will be activated during the simulation at a specified time.

The External Adaptive Traffic Control System can modify this execution by means of different actions, such as changing the duration of a phase or directly jumping from one phase to another. This is done through the GETRAM Extensions Module.

During simulation the traffic control plan structure cannot be modified (i.e. the definition of signal groups), but it is possible to change the allocation of signal groups to phases or the duration of any phase.

The Traffic Signal Control modelling is implemented using an Event Scheduling approach. At the beginning of the simulation the Control State is initialised for all signalised junctions and the first phase-changing events are scheduled.

During simulation, the control events list is revised at the beginning of every simulation step to check whether there is any change of phase due to occur during the current step in order to update the Control State.

Ramp Metering

AIMSUN2 incorporates ramp-metering control. This type of control is used to limit the input flow to certain roads or freeways in order to maintain certain smooth traffic conditions. The objective is to make sure that the entrance demand never surpasses the capacity of the main road. Ramp metering objects are located at the downstream end of a section approaching a node type juncture and affect all the lanes of the section. Figure 6 shows a ramp-metering layout.

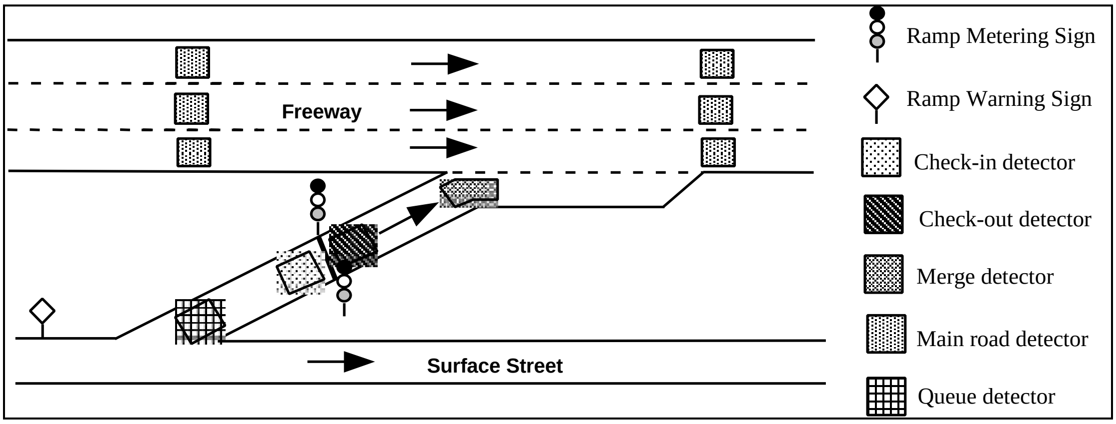


Figure 6: General layout of traffic-responsive entrance ramp metering system

AIMSUN2 considers three types of ramp metering depending on the implementation and the parameters that characterise it: green time metering and flow metering. Also there is the possibility of using the same Ramp metering model to emulate other types of access control in which the stopping time may be a given random distribution. This is delay metering.

Ramp metering objects may be located at any point of a section. Ramp metering control can be fixed, variable or adaptive. In the fixed control, the same control plan is used for the whole simulation period. In the variable control, a set of different control plans can be used at different times of simulation. Last, the adaptive control is achieved through the interfacing of AIMSUN2 to an external traffic control system. This is done through the GETRAM Extension Module.

Green Time Metering

Parameters are green time and cycle time. The ramp metering is modelled as a traffic signal that turns red and green on a cyclic basis. If it is a fixed traffic control, only a constant green time is used. In the case of simulation with some external Adaptive Traffic Control System, there would be a minimum and maximum value for the acceptable range of green time variation. The rest of the cycle time, the traffic signal will be red. Vehicles will stop at a red signal and cross at a green signal.

Flow Metering

Parameters are platoon length and flow (veh/h). The meter is automatically regulated in order to permit the entrance of certain maximum number of vehicles per hour. In this case the ramp-metering objective is to let a certain number of vehicles per hour to cross the meter. Each time the meter is opened to release vehicles, it is done in a such a way that platoons of a given length can pass. This can be done in two ways, either by counting the vehicles crossing the meter or by allocating a green time as a function of the platoon length. On average, a certain number of vehicles per hour will be released. In the case of simulation with some external Adaptive Traffic Control System, there would be minimum and maximum values for the acceptable range of flow variation.

Delay Metering

Parameters are the mean delay time and the standard deviation. This type of metering may be used to model the stop of vehicles due to some control facility, such as tolls, customs, checkpoints or any other type of individual control. It is assumed that every vehicle will have to stop at the control point (i.e. the ramp metering stop line) for a certain amount of time. This time is a random variable distributed according to a given probability distribution, e.g. a normal distribution with a given mean and standard deviation.

The ramp metering model is able to reproduce the metering control process, the behaviour of traffic at the presence of the ramp metering, and the vehicle detectors used (vehicle detection model is not described here).

The vehicle stop at the ramp metering line may be achieved by putting a dummy vehicle at the stop line which will be stopped while the ramp metering is closed and will be removed when it is opened.

The Metering Control modelling is implemented using an Event Scheduling approach. At beginning of simulation, the metering state is initialised for all controlled meterings and events corresponding to first changes of state are scheduled.

During simulation, the events list is revised at the beginning of every simulation step to check whether there is any change of state due to occur during the current step in order to update the metering state.

Variable Message Signs

Information to drivers is considered as a possible result of the actuation of a Traffic Management System on a network containing Variable Message Signs (VMS) equipment. Messages may inform the drivers about the presence of incidents, congestion or suggest alternative routes. They can even be used to make some prohibitions. AIMSUN2 takes into account the modelling of VMS and their influence on the driver’s behaviour.

Each VMS has a set of acceptable messages, and each message has a list of Actions associated with it, which represent the influence the message has on the driver’s behaviour. Upon activating a message, the associated actions are implemented. The types of message that can be modelled include modifications to the speed limits, recommendations of alternative routes and information on congestion or incidents.

An Action represents the impact that a message has on the driver's behaviour. Different types of actions are considered depending on whether the simulation is run using the Traffic Result option (Input flows and turning proportion) or the Route Based O/D Matrix simulation mode.

A Traffic Management System that displays messages on the Variable Message Signs can be interfaced to AIMSUN2 through the GETRAM Extensions Module.

Actions for a Traffic Result based Simulation

When the simulation is done using the Traffic Result option (Input flows and turning proportion) three types of actions can be defined: modifications of the speed limit, modifications of the input flow and modifications of the turning proportions.

1. *Modifications of the Speed Limit*: a new speed limit for a set of sections can be defined.
2. *Modification of the Input Flow*: an increment or decrement of the input flow can be defined as a percentage of the current flow. Input flow modifications may only affect to input sections, where traffic is generated and injected into the network.
3. *Modifications of the Turning Proportions*: the user can define an increment or decrement to the proportion of vehicles that having entered a section through an entrance will follow a turning. This is defined as a percentage of increment or reduction over the current turning probability.

Actions for a Route Based Simulation

When the simulation is Route based (using OD matrices and route choice models) two types of actions can be defined: modifications of the speed limit and Re-routing actions, which can be either modifications to the destination centroid or modifications to the next turn to make.

1. *Modifications of the Speed Limit*: works in the same way as in the Result Based mode.
2. *Re-routing Actions*: Re-routing means the possibility of altering the vehicle's path. This effect is accomplished by defining the next turn and/or defining a new destination.

    - The first type of re-routing action is the modification of the destination centroid. The user may define a set of pairs composed of the previous destination centroid and the new destination centroid.
    - The second type of re-routing action is the modification of the next turning. The user may choose among All or Selected Destinations.

    The re-routing effect can be defined by each vehicle type independently or be the same for all vehicle types. There is a *Compliance* parameter (δ) which gives the compliance level of the action, i.e. the percentage of vehicles accepting the recommendation. It can be Compulsory, Warning or Information. Compulsory means $\delta=1$, which implies that the re-routing will be followed by everybody (i.e. an obligation). Information means $\delta=0$, where the action's success will depend on the driver's behaviour (Guidance acceptance λ, a vehicle attribute). In the Warning option the user may define δ ($0<\delta<1$), which is the level of acceptance, i.e. is an advice.

Dynamic Route Guidance

We consider here Individual Route Guidance as a function of Traffic Management, whose purpose is to operate on the individual basis, guiding a specific subset of vehicles, that are supposed to be equipped, towards their destinations.

Route Guidance is only implemented in AIMSUN2 whenever the simulation is based on O/D matrices and shortest paths, which is called the Route Based simulation model. In this model, vehicles are fed into the network according to the demand data defined by an O/D matrix and they drive along the network following a given path in order to reach their destination.

During the simulation, the computation of shortest routes is determined at certain time steps. This is usually in a periodic manner, with a period that depends on the length of the section and on the level of congestion.

The simulator needs to store shortest routes from the beginning of every section to all destinations for each vehicle type at each time interval. One needs to keep all previously generated routes as long as there are vehicles using them. For each destination and instant in time, the routes are stored as a tree that makes it possible to determine how to reach the destination from any section of the network. We also attach to this tree a field that counts the number of vehicles using it. When this counter is empty, the tree may be deleted.

The procedure that we use to compute the shortest routes to a destination (either a centroid node or a section) uses a network where an arc, connecting two nodes, models a section. A special arc connecting the beginning of the turning to its end models a turning movement. The computation of shortest routes uses a label setting method, where the labels are associated with an arc. The network is constructed only once, before the start of the simulation.

The shortest route routine is a variation of Dijkstra's label setting algorithm. It gives the shortest routes from the start of every section to all destinations. The cost labels are attached to sections instead of nodes, as is usual. The arc candidate list is stored as a heap data structure. During each iteration of the algorithm, the section with minimum value is removed from the heap and the heap is restored by using efficient operations. As a new section is reached, one adds it to the heap in the correct position.

Cost Functions

Two types of section cost functions are used for calculating the shortest path trees, depending on whether or not there are simulated data available to be used for. These are the Initial Cost Function and the Current Cost Function. In both cases, the cost function represents section travel time in seconds, including the penalty of the turning movement, if it exists.

The *Initial Cost Function* is applied at the beginning of the simulation when there is no simulated data gathered to calculate the travel times. In this case, the cost of each section is calculated as a function of the travel time in free flow conditions and the capacity of the section.

The initial cost of each section, *IniCost(s),* is calculated as follows:

$$IniCost(s) = TravelTFF(s) + TravelTFF(s) \times j \times \left(1 - \frac{Capacity(s)}{MaxCapacity}\right)$$

where:

> *TravelTFF(s)* is the travel time, in seconds, of section *s* in free flow conditions. It is calculated as *Length(s)/SpeedLimit(s).*
>
> *Capacity(s)* is the capacity of section *s*, in vehicles per hour.
>
> *MaxCapacity* is the maximum capacity of any section in the network.
>
> *j :* Capacity weight. It is a user-defined parameter that allows the user to control the influence that the section capacity has in the cost in relation with the travel time.

The *Current Cost Function* can only be applied when there is some simulated travel time data available, and therefore it cannot be used at the beginning of the simulation but only when the simulation has already started and some statistical data has been gathered.

The current cost for each section, *CurrCost(s),* is the mean travel time, in seconds, for all simulated vehicles that have crossed the section during the last statistical gathering period (*TravelTime(s)).* As there may be situations in which any vehicle has not crossed a section, the following algorithm is applied to calculate *CurrCost(s):*

```
if (Flow(s) > 0) then
    CurrCost(s) = TravelTime(s)
else
    if (there is any vehicle stopped) then
        CurrCost(s) = Maximum (AvgTimeIn, IniCost(s))
    else
        CurrCost(s) = IniCost(s)
    endif
endif
```

According to this algorithm, when some vehicle has crossed the section during the last statistical period (*Flow(s)* > 0), the cost is the simulated mean travel time. In the case that no vehicle has crossed the section we distinguish the case of a totally congested section from the case of an empty section. In the first case, the cost is calculated as the maximum between the Initial Cost and the average waiting time for the vehicles in front of the queue in the section (*AvgTimeIn*). In the second case, the cost is taken as the initial cost.

Fixed Routes Mode

In the Fixed Routes Mode, shortest path trees are calculated from every section to every destination centroid at the beginning of the simulation. Then, during the simulation, vehicles are generated at origin centroids and assigned to the shortest route to their destination centroid. There is no need for a Route Choice Model as there are no alternative routes. No new routes are recomputed during simulation; therefore all vehicles always follow the shortest path and no decisions about changing to another path can be made during the trip.

Variable Routes Mode

In the Variable Routes Mode the simulation process includes an initial computation of shortest routes going from every section to every destination, a shortest route component which calculates periodically the new shortest routes according to the new travel times provided by the simulator, and a route selection model.

The simulation procedure can be characterised as follows:

1. Calculate initial shortest routes, taking as costs the estimated travel times for each section (i.e. length of section / speed limit).
2. Simulate for a period (e.g. 5 minutes) using available routes information and obtain new average travel times as a result of the simulation.
3. Recalculate shortest routes, taking into account the new travel times.
4. Add the new information calculated in 3 to the knowledge of the drivers.
5. Go to step 2.

At the beginning of the simulation, shortest path trees are calculated from every section to each destination centroid, taking as section costs the Initial Cost Function. During simulation, new routes are recomputed every time interval taking as section costs the simulated travel times obtained for each section during the last interval, this is the Current Cost Function explained before. Figure 7 illustrates when are the shortest paths (SP) calculated along the simulation period and what cost functions are used.

The user may define the time interval for recalculation of paths and the maximum number of path trees that they wish to maintain during the simulation. When the maximum number of path trees (K) is reached, the oldest paths will be removed as soon as no vehicle is following them. It is assumed that vehicles only choose among the most recent K path trees, therefore, the oldest ones will become obsolete and disused.

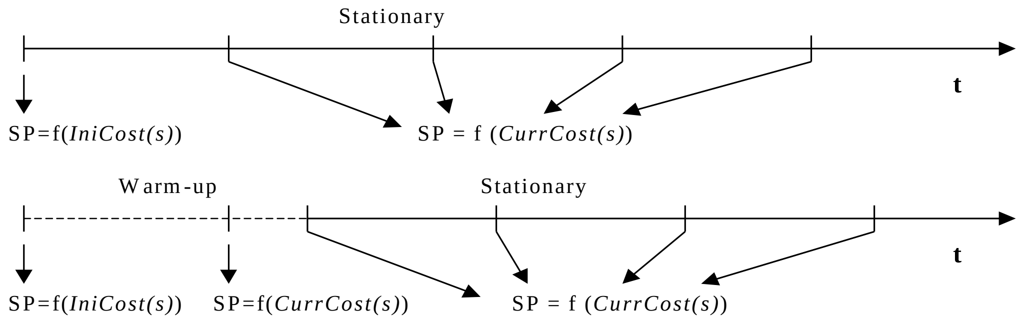


Figure 7: Calculation of Shortest Paths in a Variable Routes Mode

Static versus Dynamic Route assignment Models

Vehicles are initially assigned to a route from a set of available routes in a probabilistic way. Apart from the initial assignment of route, which is made when the vehicles departs, there is the possibility of route reassignment during the trip. This is called the dynamic route choice model.

In the dynamic route choice model a guided vehicle can make a new decision about what route to follow at any time during their trip, whenever there are new shortest routes available. In the static model, a vehicle will always follow its initially selected route to its destination, although a new shortest route could be available during the trip. Note that in the dynamic model only guided vehicles can make a decision to change to a new shortest route during the trip, as it is supposed that information is only available for equipped vehicles. Regarding this, there is a parameter for each vehicle type that gives the percentage of guided vehicles.

The behaviour of the driver in response to information acquisition may be modelled in different ways using any of the following route choice models.

Route Choice Models

Currently there are two Route Choice Models implemented, which are used either when assigning the initial path for a vehicle at the beginning of its trip or when having to decide whether or not to change path en-route in the dynamic modelling.

- Binomial Model

A Binomial (k-1, p) distribution is taken to find the probability of selecting each path. Parameter k is the number of available paths and p is the “success” probability. This model does not consider the travel costs in the decision process, but only the time at which the path was calculated. Selecting a small p will mean that oldest paths will be more likely used while selecting high values of p, the most recent paths will be more frequently taken.

- Multinomial Logit Model

We assume that the utility U_k^{rs} of route k between origin r and destination s is given by:

$$U_k^{rs} = -q t_k^{rs} + e_k^{rs}$$

Where:

q is a shape or scale factor parameter

t_k^{rs} is the expected travel time on route k from r to s, calculated as the sum of the current costs of all the sections composing the path (CurrCost(s) function as explained above), and

e_k^{rs} is a random term

The underlying modelling hypothesis is that random terms e_k^{rs} are independent identically distributed GUMBEL variates. Under these conditions the probability of choosing route k amongst all alternative routes from r to s is given by the logistic distribution:

$$P_k^{rs} = \frac{e^{-qt_k^{rs}}}{\sum_l e^{-qt_l^{rs}}} = \frac{1}{1 + \sum_{l \neq k} e^{-q(t_l^{rs} - t_k^{rs})}}$$

The scale factor q plays a twofold role making independent of the measurement units the decision based on differences between utilities, and influencing the standard error of the distribution of expected travel times:

$$Var(t_k^{rs}) = \frac{p^2}{6q^2}$$

that is:

$q < 1$ high perception of the variance, in other words a trend to utilise many alternative routes

$q >$ 1 alternative choices are concentrated in very few routes

The parameter, or scale factor q in AIMSUN2 is a user defined parameter that can be used to adjust the effect that small changes in the travel times may have on the driver's decisions.

Results Analysis Tool

A simulation model does not provide a unique solution to a given problem, it just tries to emulate the behaviour of a complex system in which randomness is involved. Each run of a simulation program, called a replication, produces a possible behaviour of the modelled system, which is a point in a sample of feasible results of the model. The final result is obtained through the statistical processing of the simulation results coming from different replications. Therefore, a simulation study requires the run of a number of replications of the same model, using different random seeds.

For this purpose, a more flexible mechanism for storing simulation outputs has been included. The user may decide to store the simulation outputs (statistics and detection) either as ASCII files or as a database, the latter using an ODBC format. In both cases the user may select where to locate these data, which makes it possible to store the results of different runs of the same model.

The idea of Experiment has been included in AIMSUN2. An experiment consists of a set of replications of the same scenario, composed of the traffic network, traffic demand, traffic control plan and a set of global modelling parameters. The user can define the number of replications to perform and the seed for each replication. Then the whole experiment can be run in Batch mode and the results of each replication are stored.

A graphical representation of simulation results is provided. Through the AIMSUN2 graphical interface the user can get time plots of different traffic variables, and also colour the network with a range of colours representing different values of the traffic parameters.

Apart from the experiment definition and storing module, the Results Analysis Tool can be completed with the addition of two further components:

- Statistical tools for result analysis: mean and variance estimation, calculation of confidence intervals.
- Statistical tools for model validation: hypothesis test, regression analysis.

The outputs provided by this module are both the simulation statistical output data and the simulated detection data. This data can be stored either in a database or in ASCII files. In the first case, the different results from each replication are stored using different primary keys, while in the second case the results from each replication are stored in different subdirectories.

DRACULA

Introduction

In order to comply with the Model Update Specifications proposed in SMARTEST Deliverable 4, the following five models have been improved within DRACULA:

- Roundabouts,
- PT Services,
- Adaptive Traffic Signals,
- PT Priority,
- Detectors

and one new model has been added:

- Traffic Calming.

Improved validation of the model has also taken place. Improvements in the PT services model include a new bus stop model and the development of guided bus and tram operations. New roundabout and traffic calming models have also been developed, which have been calibrated and validated using data collected in Leeds.

The adaptive traffic signals improvements concentrated on linking DRACULA to a BALANCE UTC system that is due to be installed in Leeds and Sheffield. The installed BALANCE system will use the TCP/IP communications protocol to link up its various components. With this in mind a DRACULA interface that also uses TCP/IP has been developed. The improvements in the detector model in DRACULA concentrated on providing the BALANCE system with the on-street information it required. As well as the usual loop detector data this also included both public transport and emergency vehicle location information.

Roundabouts

DRACULA makes the following assumptions about each roundabout:

- the roundabout is circular,
- the roundabout is modelled as a continuous link, with a given number of lanes, with entry and exit points at positions along it,
- vehicles attempt to travel at a desired circulation speed when on the roundabout,
- the usual car following rule is used on the roundabout,
- a new lane changing rule is used on the roundabout.

When each vehicle is generated it is given a scale factor to use when calculating its desired speed on a link. This scale factor is based on a random selection from a distribution with a given mean and variance. Each vehicle type has a given mean and variance to use.

The roundabout model uses three regimes. Firstly on approaching the roundabout vehicles have to get into an appropriate lane. When vehicles arrive at the roundabout they have to determine whether there is a suitable gap to allow them to enter the roundabout. Finally, when travelling on the roundabout vehicles have to choose an appropriate lane to allow them to leave the roundabout at the desired exit.

Public Transport Services

The main improvements to the Public Transport service models within DRACULA have been:

- a new public transport service model
- a new public transport stop model

- a new reserved lane model
- guided bus operation
- Public transport service

The public transport vehicles enter the network at regular service frequency. They follow the pre-defined fixed route through the network as other traffic except when using reserved public transport lanes where provided, stopping at public transport stops for the service.

Public transport stops

There are two elements required to model public transport vehicle motion in the vicinity of public transport stops. Firstly, when approaching the stop, public transport vehicles need to move into the lane where they can access the stop. The public transport vehicles begin to attempt this manoeuvre in the link before the link with the stop. Secondly, if there are passengers waiting at the stop then the public transport vehicle has to stop at the stop for sufficient time to pick up all the passengers.

A test network in Leeds has been used to calibrate and validate the bus stop model. Data has been collected, using moving observers, on the journey times of buses between five bus stops and dwell times of buses at these stops for buses travelling down the Scott Hall Road between Potternewton Lane and Sackville Street during the morning peak period. A summary of this data is presented in Table 14 and Table 15. The mean value, the number of observations (N) and the standard deviations (s.d.) are given. The upper and lower limits of the confidence interval, at the 95% confidence level, between which it is expected that the mean value will lie are also given in the tables.

Stops	Mean (s)	N	s.d.	Lower (s)	Upper (s)	DRACULA (s)
1-2	21.88	33	4.285	20.42	23.34	24.9
2-3	36.31	16	5.654	33.54	39.08	38.9
3-4	40.87	30	15.900	35.18	46.56	34.6
4-5	39.92	49	12.670	36.37	43.47	44.4

Table 14: Bus journey times between stops during the morning peak (08:00-09:00)

Stop ID	Mean (s)	N	DRACULA (s)
1	25.9	31	25.7
2	19.8	12	19.7
3	38.4	25	40.7
4	22.6	21	23.9
5	11.6	23	13.5

Table 15: Dwell times at stops during the morning peak (08:00-09:00)

The final column in the tables shows the value output from DRACULA for these times. Mean values from five simulation runs were calculated. As can be seen there is good agreement between the observed and the modelled journey times and bus stop dwell times.

Reserved public transport lane

The following pseudo code describes the movement of public transport vehicles as they approach and move into a reserved lane.

```
if a reserved public transport lane is in the next link then
    try to change to the lane in the current link which leads to the reserved lane
    if failed to change lane then
        stay in lane until the next link
    end if
end if
```

Once in the link with the reserved lane the following logic applies

```
if there is a reserved public transport lane in the current link then
    try and move into the reserved lane
end if
if the reserved lane permits the public transport vehicle’s next turn then
    stay in the reserved lane
else
    move off the reserved lane into a lane that allows the turn, when near the junction
end if
```

Guided bus operation

The operational distinction between a guideway and a reserved lane which this implementation incorporates is that a bus may join the guideway only at dedicated points on the route whilst a bus may “drift” into and out of a reserved lane anywhere along its extent. A lane can be specified as reserved for one particular type of vehicle or a combination of vehicle types.

Additional outputs include the public transport service route specified summary statistics, which include the total travel time, distance, average speed, fuel consumption, pollutant emissions for the service route. As the user’s request, each public transport service vehicle’s link-by-link travel times are also output.

Adaptive Traffic Signals

It is becoming increasingly common to link micro-simulation models to real urban traffic control (UTC) systems and to then let the two systems interact. The UTC systems can obtain data from the simulated network, such as vehicle detections, and use this information to perform control actions in the simulated network. This approach has considerable merits. It negates the need to produce a model to replicate the effects of the UTC system. It also allows accurate simulations to be performed without the modeller having to know precise details of the how the UTC system works. This can avoid commercially sensitive information having to be revealed to the modeller.

Within the SMARTEST project DRACULA has been linked to the BALANCE UTC system. BALANCE is a distributed UTC system that has been developed at the Technical University of Munich. It underwent field trials in Munich within the EC funded DGXIII LLAMD project, and has since been tested in three other European cities, namely London, Glasgow and Belfast. It is due to be used in Leeds and Sheffield in the near future.

Within a BALANCE system, decisions about signal settings at individual junctions are made by Micro-BALANCE outstations at each junction. Strategic control decisions at the network level, which can override or weight decisions at the junction level, are made by a centralised Macro BALANCE computer. BALANCE uses standard TCP/IP communications protocols to communicate with signal controllers on-street and between its system components.

An interface between DRACULA and BALANCE was therefore developed which used the same TCP/IP communication protocols as used by the BALANCE system on-street.

By using a multi-tasking operating system, such as Windows 95/98/NT, it is possible to run all the micro-BALANCE tasks on a single computer, if required, rather than use a separate computer for each one, as would be the case in the real world. A flexible approach was developed in the project to allow the micro and macro BALANCE tasks to be spread across a computer network as available. Similarly it should be possible to treat DRACULA as just another task and run it on any of the PCs in the computer network. In practice this caused problems if DRACULA and all the BALANCE tasks were chosen to run on a single computer. It proved difficult to display the animated graphical outputs of DRACULA simultaneously with the outputs from BALANCE. A further problem was encountered when considering how to link the standard communications routines used by BALANCE into DRACULA. The DRACULA software is currently DOS based and is incompatible with the Windows based communications routines. It therefore proved impossible to link the communications routines with DRACULA directly. This problem was overcome by writing a small Windows program, called SPRUCE, to handle the communications, which was run every simulation step within DRACULA.

The design for linking DRACULA to BALANCE was based around a further INTERFACE task that sat between DRACULA and the BALANCE tasks. The INTERFACE task acted as a server. It handled communications between all the tasks, ensured all the tasks were synchronised, and translated all the data going between the tasks into the required formats. The INTERFACE task also performed some of the duties normally carried out by on-street signal controllers, such as checking that minimum green periods had elapsed before changing signal phases.

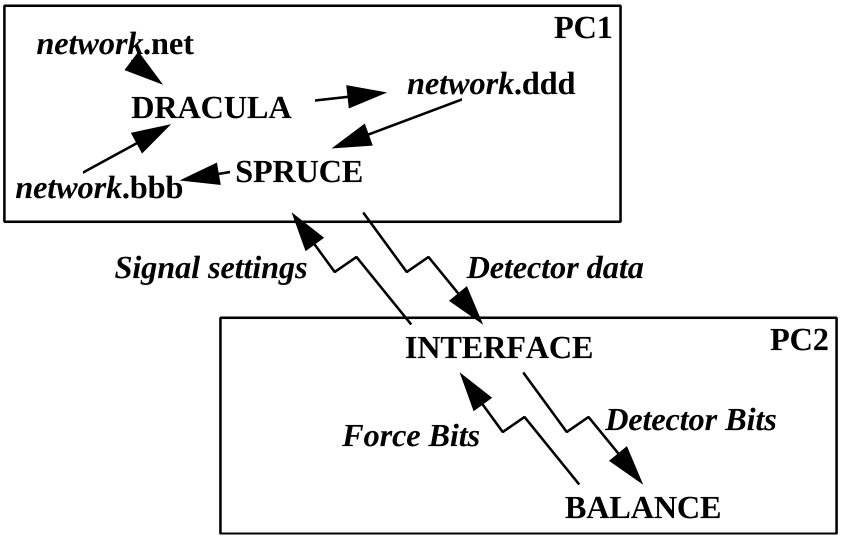


Figure 8: DRACULA - BALANCE Data Flows

The initial proposal required that DRACULA should communicate with BALANCE via an INTERFACE program. This interface is based on a function that would need to be used by BALANCE to send out signal settings and receive detector data. This function was written using Microsoft Visual C++ and used TCP/IP communications protocols. It was supplied via a DLL called ChatDll.dll and contained the exportable function 'XDataOnTCPIP' which takes four arguments as shown below

int XDataOnTCPIP(char* RemoteIP, int RemotePort, char* Msg2Send, char* Msg2Take);

This function just uses TCP/IP to send and/or receive message strings between two computers on a network. The function was to be called every second, by DRACULA, BALANCE and the INTERFACE program to transfer data between them.

To use the function it required:

i) BALANCE to be adapted to use the interface function

ii) the translation of the data going to (detector bits) and from (force bits) BALANCE into something DRACULA could understand. The INTERFACE program performed this.

It proved impossible to incorporate the XDataOnTCPIP function directly within DRACULA. Instead a simple program called SPRUCE.EXE was written which did incorporate the function. This program was run at each simulation step by DRACULA using the standard *system* function. At each simulation step, DRACULA would write detector data to a file called network.ddd and read signal settings from a file called network.bbb. The SPRUCE program would then be called and it would read the data in the network.ddd file and transmit it to the INTERFACE program and receive the signal settings back from the INTERFACE program at the same time. SPRUCE would then write the signal settings to the network.bbb file. (see Figure 8)

The INTERFACE program performed the following functions:

- Translation of the BALANCE force bits into appropriate signal settings.
- Translation of the DRACULA detector data into the stream of detector bits required by BALANCE
- Synchronisation of the tasks

BALANCE uses the detector data it receives to optimise the signal settings. The detector bits are received and the signal force bits transmitted using the XDataOnTCPIP function.

Every second, DRACULA produced a list of SCOOT nibbles for all of the detectors in the network. Every second the INTERFACE program sends detector bits to BALANCE and signal aspects to DRACULA. Every second BALANCE outputs stage force bits messages, which are full 16-bit UTC control bit pattern, but only using the stage force bits.

A single four arm junction in SW London was used to test the operation of the DRACULA / BALANCE interface. A careful check was made to ensure that the signal plans being recommended by BALANCE were being implemented in DRACULA.

Public Transport Priority

Apart from providing public transport with special reserved lanes, public transport is also given priority at signalised intersections. When a public transport vehicle is detected at time t_0 and predicted to arrive at the stopline at time t_a, one of two actions may be performed:

- *Extension*, which extends the bus green period in order to allow the bus to exit;
- *Recall,* which terminates the bus red stage earlier in order to reduce the bus waiting time.

Figure 9 shows schematically the signal priority in a space-time diagram. The signals for the bus link are shown on the top, with t_r and t_{ra} representing the start and end time of the red aspect. ra denotes red/amber. $t_{ext}=t_r+E_{max}$, where E_{max} is the user specified maximum allowed extension (in second). The distance from the detector to the stopline is d. Three bus trajectories from the detector to the stopline are drawn in dashed lines.

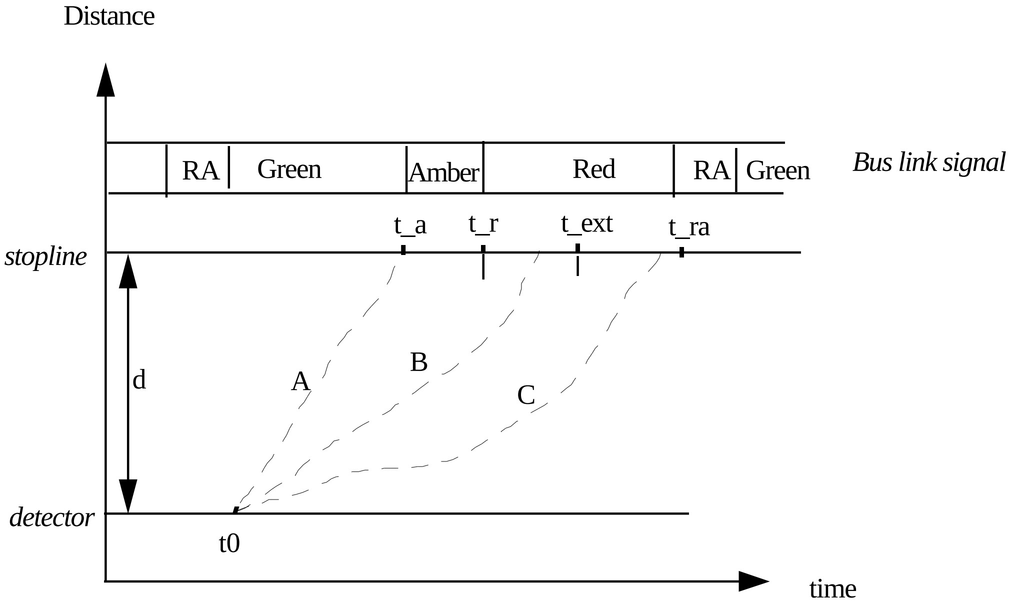


Figure 9: Space-time representation of bus signal priority

If a bus is predicted to arrive at the stopline just after the start of the red signal (case B in Figure 9), the bus green aspect will be extended by just enough time to allow the bus to exit. The amount extended depends on the predicted bus arrival time, subject to a user-defined maximum (E_{max}) and to minimum greens for the subsequent stages affected.

If a bus is predicted to arrive during the red, but an extension is not appropriate (i.e. requires more than the maximum permitted extension, case C above), then the duration of the bus red aspect may be reduced by a constant amount of 5 seconds. The length of other stages remains unchanged, so the length of the current cycle is decreased temporarily.

Otherwise the signals will not be changed (case A in Figure 9).

The operation of model has been checked using a test network in Leeds. It was not possible to validate the model as no scheme using bus priority has yet been adopted on the test site.

Detectors

Detectors in DRACULA have been modified to allow them to output the data produced by SCOOT detectors, which are common in the UK. The data consists of quarter second occupancy bits that are sent out every second as blocks of four bits.

The front and rear ends of a vehicle are compared to the location of a detector in the current lane the vehicle is travelling in, a detection is triggered if a vehicle passed or is stopped on a detector. The exact timing, in quarter second intervals, when the vehicle passed the detector is extrapolated based on the current speed the vehicle. This is because the simulation time increment is one second.

At every simulation time step the program loops through all detectors in the network and outputs all detections. The detector data is in the form of SCOOT nibbles. This is quarter second occupancy data that is sent every second. For each detector four bits of information are sent every second. The information is passed using bytes (i.e. 8 bits), so two detectors worth of data are sent with each byte. The SCOOT nibbles are created left to right, so the leftmost bit is for the first quarter second, the rightmost bit for the last quarter second.

The network used for the calibration test of Adaptive Traffic Signals was also used to check the correct operation of the detector module.

Traffic Calming

Traffic calming is represented as a special speed-controlled region in a link.

When approaching a traffic calming region:

```
If the current speed is more than the maximum speed of the region, then
    Decelerate at a normal deceleration to the maximum speed of the region
else
    Move at the car-following speed
end if
```

A simple model including a traffic calmed section was built to check the correct operation of the new model.

NEMIS

In order to comply with the Model Update Specifications proposed in WorkPackage 3, the main effort has gone into providing the micro-simulator NEMIS with an improved and standardised interface suitable for the simulation in real time of the following external Transport Telematic Applications:

- *Adaptive Traffic Signals*
- *Public Transport Priority*
- *Variable Message Signs*
- *Dynamic Route Guidance*

A standard interface, based on a TCP/IP communication protocol was adopted to connect the computer where the model runs to the network where the external strategies operate.

Furthermore, two models have been improved in the new release of the micro-simulator:

- *Public Transport Services*
- *Vehicle Detectors*

In more detail:

- Public Transport Services model has been enhanced with the introduction of layby PT stops.
- Vehicle detector improvement concerns performance, error rate and breakdown occurrence.
- The validation of the standard interface mainly deals with operational aspects. Stress conditions will be generated by connecting the model to a real network of SPOT traffic control units.

The New Interface

The calibration and setting up of a UTC system on street can be extremely time-consuming and difficult unless extensive off-line tests are performed in a controlled environment.

The NEMIS software package was designed specifically as a tool for testing urban traffic control strategies prior to or in parallel with on-street testing.

NEMIS already supports an interface with external roadside processors (e.g. SPOT or SCOOT OTU) to test the effectiveness of urban traffic control systems as well as to screen strategies and tune system parameters before field installation.

The interface package needs to be installed on a MS-DOS PC connected by a serial line with the NEMIS computer and by another serial line with a SCOOT system or to the SPOT MFOs[1] of the UTOPIA Network (Figure 10).

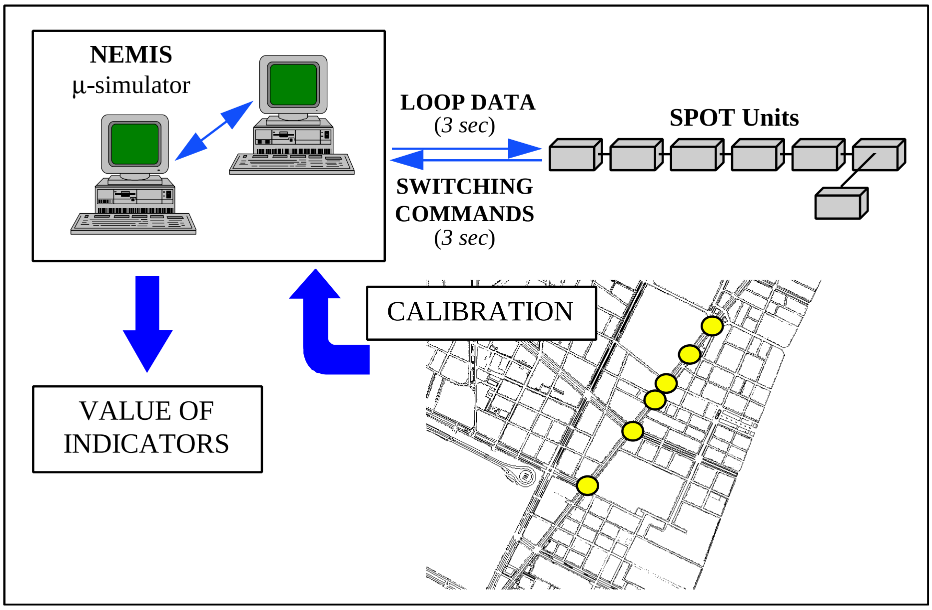


Figure 10: Use of NEMIS as a system evaluation tool before field installation

---

[1] MFO = Multi Functional Outstation

The existing interface communication protocol operates using several serial connections (RS232) between the NEMIS computer, the MS-DOS PC where the interface package runs and the real system. This solution presents the following limits:

- limited communication capabilities (limited speed of serial connections)
- non-standard communication protocol (proprietary protocol "ad hoc" to interface NEMIS with SPOT and SCOOT)
- limited simulation capabilities (dedicated hardware is needed for the intersection controller)

The new interface overcomes these limits by using a standard protocol based on TCP/IP in order to connect NEMIS with a LAN/WAN where the intersection controllers implementing external Traffic Control Strategies (e.g. MFOs SPOT) operate.

The new NEMIS interface package provides users with a simplified and highly efficient micro-simulator suitable for investigating the impact of Advanced Transport Telematics functions (adaptive and co-ordinated traffic signals, public transport priority, VMS and Dynamic Route Guidance) on large network areas.

The main objective of implemented modifications (as shown in Figure 11) is to have a new interface package based on standard communication protocol TCP/IP and suitable to directly interface NEMIS with several external control strategies embedded in UTOPIA SPOT units. In fact, the new interface package is able to manage messages written directly in the UTOPIA format.

It is also possible to use NEMIS to test any “ad-hoc” external control strategy. If this is the case, the external control strategy developed needs to exchange appropriate messages with NEMIS using the proposed format for each message.

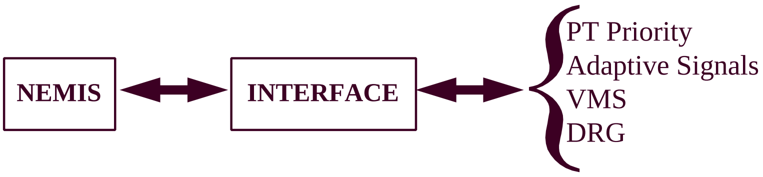


Figure 11: Target of the new interface package

Starting from the physical approach that is shown in Figure 11, the new interface package has been developed according to the architecture proposed in Figure 12.

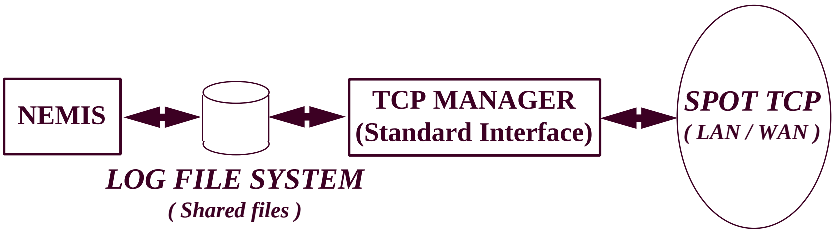


Figure 12: Implementation of the new interface package

The new interface package is composed of two parts:

• TCP MANAGER	tools that manages the communications between NEMIS and the network where the external control strategies are located
• LOG FILE SYSTEM	A set of circular files where the messages that need to be exchanged between NEMIS and the TCP MANAGER are stored.

The behaviour of the new interface package can be easily described by means of the following steps:

- The TCP MANAGER receives from the external control functions that reside within SPOT units or within others users developed packages, all the messages containing the elaborated control strategy. The communications between the TCP MANAGER and the external control functions are based on the TCP/IP standard protocol.
- When the TCP MANAGER receives a new message, the message itself is processed and then written onto the appropriated file into the Log File System.
- The TCP MANAGER also reads the circular files HIPRY and LOPRY (two command files that belong to the Log File System) where NEMIS write the messages needed by the external control functions.
- When a new message is written by NEMIS into a command file of the Log File System, the TCP MANAGER, processes the message and then sends it out towards the appropriate SPOT unit or to the appropriate external control function.

The flow chart in Figure 13, shows in a schematic way, the behaviour of the two main tasks of the TCP MANAGER and the existing interaction between NEMIS, the TCP MANAGER and the whole simulation network.

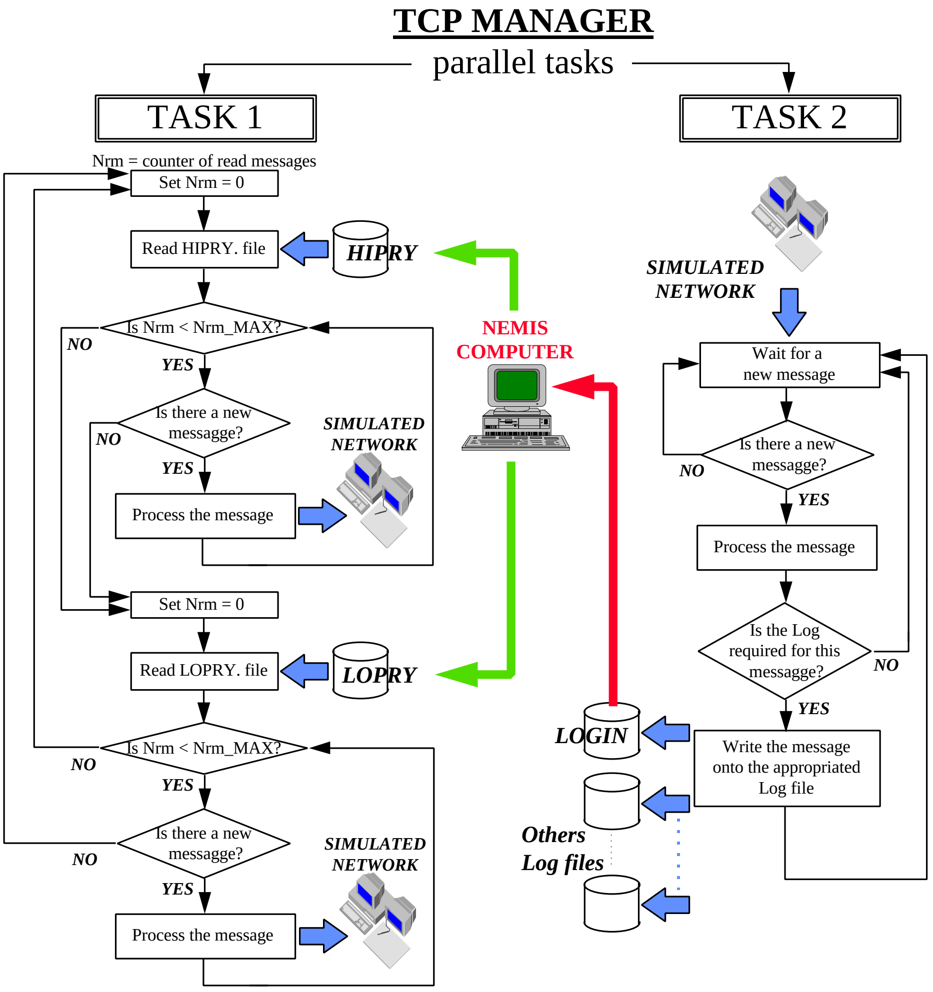


Figure 13: TCP MANAGER behaviour

The messages are written onto the Log File System using the LFS management functions provided together with the interface package. The same LFS management functions have to be used to read the messages previously written onto the Log File System by NEMIS itself and/or by any other external operating tools.

Comparing the new interface with the old one, we can highlight the following advantages of the new interface:

- it uses a standard communication protocol
- the communication speed is limited only by the communication network features (speeds greater then 10Mb/s can be achieved by using optical fibre)
- the simulation does not need dedicated hardware (a standard PC network can be used)

Public Transport Services

The micro-simulator NEMIS supports a detailed model for Public Transport management.

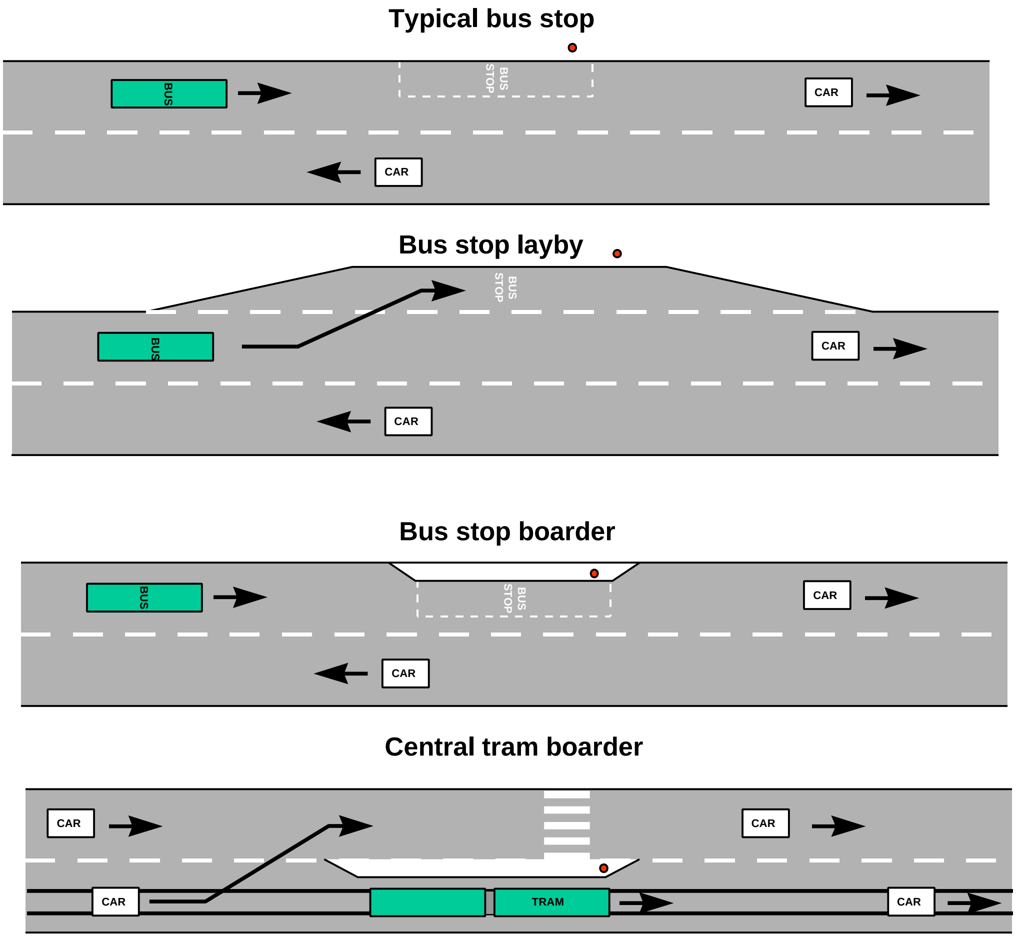
The main aspects of this model are the following:

1. PT vehicles are generated at the terminus with a random headway depending on the nominal frequency and variation defined for the service. Nominal parameters must be specified during the network coding process. During the simulation, as a PT vehicle is introduced into the network, an extraction from the distribution is made to evaluate the generation time for the next vehicle of the service. Each service has an independent random process as it has its own generation seed for the random extractions.
2. Time spent by a vehicle at a stop is randomly extracted from a distribution that changes according to the service and stop. An average stop time and a standard deviation must be specified for each stop. Another stop characteristic (that must be specified during the network coding process) is the distance of the stop from the previous one (or from the beginning of the link) in metres.
3. A PT vehicle that is moving on a link where no stops are located or that has already done all the stops on the link, can move as a private vehicle, so it could change lane and overtake other vehicles.
   If the PT vehicle has still stops to do on the current link, it stays in the lane where the next stop is, following the vehicle in front.
4. It is possible to define a separate set of traffic signals for PT vehicles that use reserved lanes. During the coding process, traffic signal for PT vehicles must be described using the same syntax as used for the “private” ones (See NEMIS User Manual, Sec. 4.4).
5. Statistics about PT are reported in an output file.

Point 2 of the previous list, highlights the characteristics of the PT stop as it is implemented in NEMIS. It clearly appears that there is no information regarding the kind of the stop itself. With reference to Deliverable 4 (Appendix A - Sec.1 “Public Transport Services”), four types of PT stops must be provided by the micro-simulator in order to model the various types of public transport stop found in road network throughout Europe. Figure 14 shows these four main types of public transport stop.

The NEMIS model does not take into account the possibility of having kerbside parking at the bus stop, furthermore no models for passenger generation are provided (time spent by a vehicle at a stop is randomly extracted from a distribution that changes according to the service and stop.). From this point of view, the behaviour of the drivers that follow a PT vehicle is the same for both a typical bus stop and bus stop boarder. If there is a lane available to overtake the bus and if the gap is suitable to change lane, the driver can change lane, overtaking the PT vehicle. If no overtaking lane is available, private vehicles can only stay in the lane where the PT vehicle is, following the vehicle itself.

Similarly, in the case of central tram boarder, the drivers that follow the PT vehicle must slow down their speed and look for a suitable gap to change lane and overtake the tram. In this case, some attention has to be paid to the passengers that are leaving the tram and that could cross the lane where private traffic is flowing. For the layby stops, when the PT vehicle reaches the stop, it leaves the lane on the carriageway and private traffic can flow normally on the lane. Later, the PT vehicle must look for a suitable gap in order to leave the stop, and, if this is the case, the private vehicles should give priority to the PT vehicle that is leaving the stop


Typical bus stop

Bus stop layby

Bus stop boarder

Central tram boarder

Figure 14: The four different types of public transport stop (driving on the left)

The micro-simulator NEMIS has been enhanced with the introduction of two different kinds of PT stops:

- Typical Bus Stop : suitable for simulating typical bus stops, bus stop boarders and central tram boarder (this last assumption is true if we do not take care of pedestrians that can cross the lane next to the stop, and that can slow or stop the incoming private vehicles)
- Bus Stop Layby : suitable to simulate bus stop laybys.

The Public Transport stop model is able to reproduce the presence of a PT stop in the simulated network and the behaviour of traffic within the region close to the stop.

When a PT vehicle enters a link, its behaviour depends on the presence of possible stops within the link.

If no stops are forecast for the PT vehicle on the link it has entered, it can move freely as if it were a private vehicle, using all the lanes available on the link and changing lanes to overtake any vehicles that proceed slowly ahead of it.

If the PT vehicle must stop on the link, it must change lane to the one where the stop is located, so the lane changing procedure for PT vehicle is actuated. When the PT vehicle has achieved the correct lane, it proceeds, following the vehicle in front until the stop has been reached. In order to avoid the situation where a PT vehicle stops in the right position but in the wrong lane during the lane changing process, vehicles proceeding on a parallel lane should give priority to the PT vehicle when it has shown its intention to change lane.

Before the stop is reached, a random extraction of the stop time will be produced, using the average stop time and the standard deviation produced as input for the model.

When the PT vehicle reaches the stop, its behaviour and the behaviour of any following vehicles, depends on the kind of stop. As said in the input section, two kinds of stop are provided: normal (grouping the typical bus stop, the bus stop boarder and the central tram boarder) and layby.

If the PT stop is of the normal kind, the stopped PT vehicle blocks the lane during all the stop time so that the following vehicles must change lanes to overtake the stopped PT vehicle. For the central tram border stop, the vehicles that pass the stationary tram on the inside lane should give priority to any passengers that have left the tram and that will cross the lane using the pedestrian crossing which is always provided for this purpose. To model this feature requires the development of a passenger model or, alternatively, the determination of the number of passengers that have left the PT vehicles and want to cross the inside lane. This can be modelled on the basis of a probabilistic distribution based on the stop time and on the average width of the inside lane.

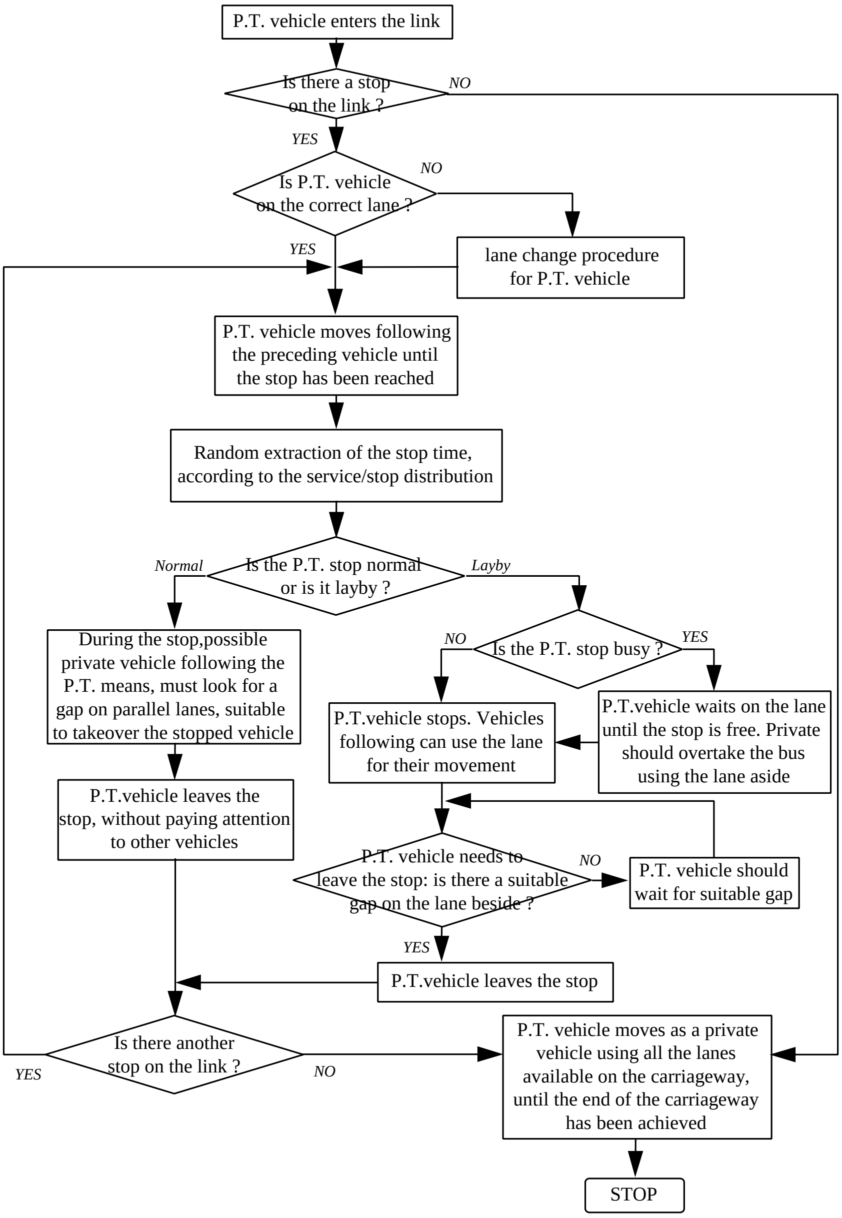


Figure 15: Public transport stops processing

In the case of a layby PT stop, the stopped PT vehicle leaves the lane where it was proceeding so that, during all the stop time, the private vehicles following the PT vehicle can proceed normally, without changing lane to overtake the stopped PT vehicle.

Of course, the layby PT stop can serve different services so that, when the PT vehicle reaches the stop, a PT vehicle of a different service might already occupy it. In this case, the dimension of the layby area (expressed in number of buses that can occupy the layby area simultaneously) determines whether the stop can be occupied immediately or whether a waiting period is required until the stop is free. If there is no space available in the layby area, the PT vehicle should stop on the lane waiting for a slot and thus block the lane to all following vehicles that must now change lane to overtake the stopped PT vehicle.

Furthermore, although the PT vehicle has priority during its lane changing movements, when it needs to leave a layby stop area, it must look for a suitable gap between incoming vehicles on the closest lane before moving. In any case, approaching private vehicles should give priority to the PT vehicle.

Figure 15 shows all these procedures in a schematic way.

Detectors

In the context of the models specifications (See Del.4 - Appendix A Sec 7), a definition has been carried out for a general detector, that is a device that provides measurements of variables that have to be selected by the users. In this case the detector technology does not matter and the interest for micro-simulation models lies in the data that can be measured and exchanged.

Another definition has been carried out in order to classify detectors into two different classes: passive detectors (take variable measurements and do not exchange data with vehicles) and active detectors (take variable measurements or receive information from vehicles and can also send information (Dynamic Route Guidance practice) to vehicles).

In NEMIS, the capability to simulate detectors is fundamental for a correct application of Adaptive Traffic Signals and PT Priority strategies[2], so there is a strong interest in the various types of passive detectors.

In the current state of the art models, the detectors are placed within the network during the set up procedure and they detect the passage of a vehicle when they reach the point where detectors are placed. No technical fault or breakdown capabilities are provided for the sensors.

Furthermore, in some cases, the users could want to test the robustness of the applied strategies when a breakdown occurs in one station of detectors or when a detector counts too much or too little. In this new enhanced release of NEMIS, it is possible to define a bias percentage for the sensor counts during the loading procedure.

Hence, the bias percentage is a static parameter that cannot be varied during the simulation time. Different simulation issues should be executed with different bias percentages for the same set of sensors in order to compare the behaviour of the external control strategies with different operative scenarios.

The detectors are placed across the carriageway (one detector for each carriageway) in a specified position.

Their behaviour could be summarised as follows:

- the routine that manage the vehicle movement moves the vehicle
- if there is a detector on the carriageway and if the examined vehicles has overtaken the detector, the detector counts the vehicle

---

[2] Detectors for the PT locator are already provided by the simulator.

- if a bias percentage not equal to 1 is defined for the detector itself, the count is update multiplying the last count for the bias factor
- the routine that manage the vehicle movement moves next vehicle

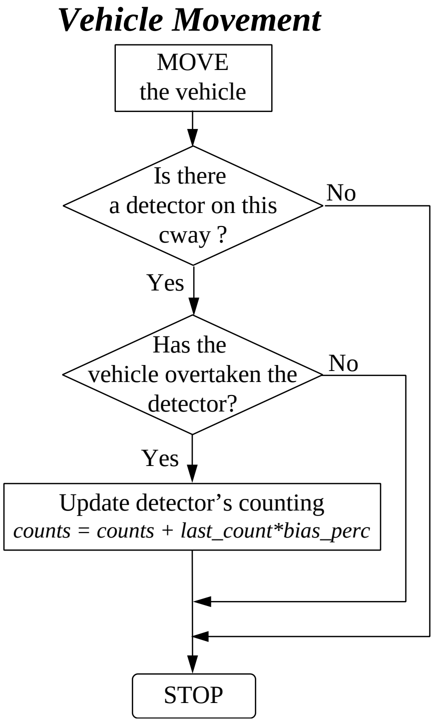


Figure 16: Detectors processing

Figure 16 shows a schematic diagram of the detectors processing procedure.

The outputs of the detectors processing are the traffic counts that will be directly use by the simulator to produce the input messages required by the external control strategies.

Adaptive Traffic Signals And Public Transport Priority

Adaptive control strategies for private traffic and public transport priority services, are key elements of urban traffic control systems.

Taking into consideration the growing interest shown by users for simulating the main telematic functions for traffic management with micro-simulation, the next generation of micro-simulators must provide users with the capability of simulating these fundamental functions.

It is also clear that the state of the art technology proposed for the implementation of adaptive control and public transport priority strategies is far from being standardised.

Hence, it is a good idea to separate the micro-simulator itself from the software module implementing the control strategies.

This last consideration gave birth to the idea of providing users of the NEMIS micro-simulator with a tool able to interface, directly and in an easy way with the external control strategies embedded in the SPOT unit (local multifunctional unit of the UTOPIA integrated system). The adopted approach lets NEMIS, in the easiest possible way, have the capability of evaluating the impact of particular adaptive control and public transport priority strategies (those implemented by SPOT unit). It also allows the simulation of control strategies developed ad-hoc by the user.

There follows a brief introduction of the UTOPIA integrated system, and of the adaptive control and public transport priority strategies implemented by the SPOT units at the local level.

UTOPIA (Urban Traffic OPtimisation by Integrated Automation) is a system designed to improve urban travel conditions by the application of fully automated control principles. The concept is based on a specific system architecture and control strategy. It was conceived as an innovative response to two fundamental requirements of wide-area traffic control systems:

- significant improvements in private vehicle mobility in all traffic conditions
- the assignment of absolute priority to selected public transport vehicles at traffic signal intersections

UTOPIA control strategies aim to reduce significantly the total time lost by private vehicles during their trips within the controlled area, subject to the constraint that public vehicles requiring priority shall not be stopped at intersections with traffic signals.

Field trials have shown that systems based on the UTOPIA concept can meet both the above requirements simultaneously[3].

The UTOPIA architecture is hierarchical and decentralised: optimal control strategies are determined at the higher level on the basis of area traffic prediction, while traffic signal control is actuated at the lower level according to the actual traffic conditions encountered at the intersections.

UTOPIA control strategies overcome the unmanageable complexity of the area optimal control problem by a process of decomposition into several simpler, interrelated sub-problems. Control problem decomposition, the choice of suitable functions for the resulting sub-problems and the introduction of rules for sub-problem interrelation allow a hierarchical and decentralised architectural solution to the original control problem and provide control actuation that is close to optimal. Decomposition is performed by following a topological rule: firstly, the area is subdivided into ‘overlapping’ zones, where each zone is centred on a controlled intersection and includes neighbouring intersections. Then an optimal control problem is defined for each zone.

The zone control problem concerns the traffic control to be actuated at the central intersection only, but it provides a consistent interrelation with the control of the neighbouring intersections and takes into account traffic information and traffic control data concerning the whole zone. The function to be optimised consists of the terms relating to the traffic observed on the incoming link of the central intersection and those that implement the following two fundamental interaction principles:

- a *strong interaction principle* is implemented by taking into account the time lost at the downstream intersections by vehicles leaving the central one;

- a *look-ahead principle* is implemented by taking into account traffic forecasts defined on the whole optimisation horizon (120 sec) for all the central intersection’s incoming links.

In order to guarantee stability and robustness at the network level, interactions are provided with a higher level, where an area optimal control problem is defined on the basis of the area traffic macroscopic model.

The final result is a series of problems that can be classified as belonging to two different classes: the ‘intersection level’ and the ‘area level’. At both levels the problem formulation is based on the definition of two fundamental modules: the *state observer* and the *controller*.

Control problems are solved by using a network of interconnected control units that apply suitable Open Loop Feedback Optimal (OLFO) techniques.

The new interface approach developed for NEMIS is based on the following components:

• the LogFile System	set of circular files that are used to store the messages exchanged between NEMIS, the TCP Manager and the SPOT

---

[3] UTOPIA - Technical Reference Manual

	units' network
• the Library for the Log File System management	Library that contains the FORTRAN functions used by NEMIS in order to manage the access to the circular files set, and the operation of reading/writing messages onto circular files
• the TCP MANAGER	a front-end TCP that, manages the communication interface of the whole UTOPIA system and the data exchange between NEMIS and the local controller unit (SPOT) of the simulated network

Adaptive control and PT priority are external strategies; i.e. external tasks embedded within the local SPOT unit of the UTOPIA System.

In this section, instead of a description of the strategies themselves, the new interface provided, that allow users to implement the information exchange between the external modules where the control strategies reside and the NEMIS micro simulator is described.

- The circular files

Figure 17 shows the interface between the SPOT units (unit where the software module that implement the external adaptive control and public transport priority strategies resides) and the micro-simulator NEMIS.

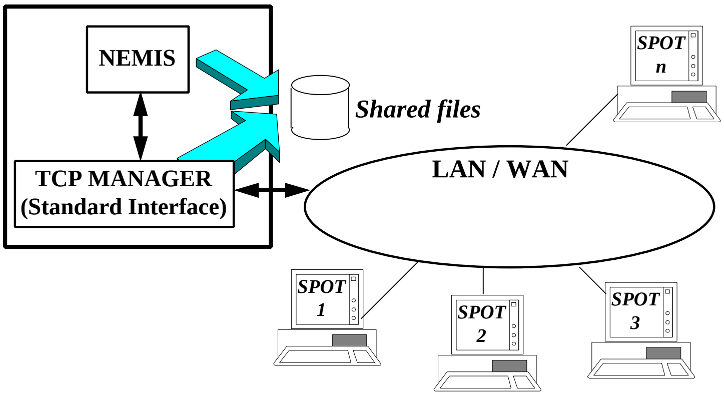


Figure 17: Scheme of the new interface

The communication takes place by means of messages logged by the TCP Manager onto a set of circular files: the Log File System. The term "circular files" refers to a set of logical structures that allow sequential accesses, in which the first record logically follows the last one. Circular files are used to store records of the messages that are exchanged between the various elements of the whole simulation system. They contain all the messages produced by NEMIS, by the local SPOT units and by possibly other tools of the UTOPIA system (users interface tools).

Within circular files, records are sorted on the basis of the date and time of the logged messages.

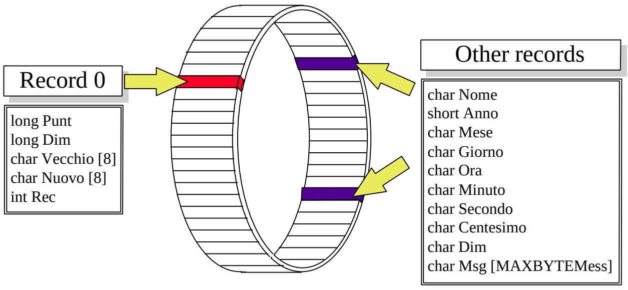


Figure 18: Circular file elements

Within the Log File System, NEMIS and the TCP manager exchange information using the three following circular files:

• HIPRY.	HIgh PRioritY messages	contains messages 78 (PT forecasts) written by NEMIS. The TCP Manager reads the records in this file with a user defined time period (default = 0.5 sec)
• LOPRY.	LOw PRioritY messages	contains messages 93 (detector counts) written by NEMIS the TCP Manager reads the records written in this file with a user defined time period (default = 0.5 sec)
• LOGIN.	LOG INput messages	contains messages 91 (“planned” signal plan) and messages 111 (synchronisation messages) written from the TCP Manager. NEMIS reads this file searching for synchronisation messages (MSG 111) and “planned” signals plan messages (MSG 91)

- The exchanged messages and their format

The messages needed to implement the simulation of adaptive control and public transport priority strategies are the following:

- MESSAGE 78: PUBLIC TRANSPORT FORECASTS (for PT PRIORITY)

  Communicates to the external control strategy, the arrival time forecast for the PT vehicle. It can also contain data related to the travel time between detector and stop line.

  It is logged onto the circular file “HIPRY.” of the Log File System

- MESSAGE 93: PRIVATE TRAFFIC DETECTORS (for ADAPTIVE TRAFFIC SIGNALS)

  Communicates data related to the traffic counts of the detectors directly connected to the micro.

  It is logged in the circular file “LOPRY.” of the Log File System

- MESSAGE 91: STAGE PLANNED (for ADAPTIVE TRAFFIC SIGNALS / PT PRIORITY)

  This message is sent out to the upstream intersection (that needs to know the future strategy planned by the downstream intersections in order to achieve the strong interaction principle) and to NEMIS. NEMIS can then vary the SEM6 matrix (matrix for the traffic signal description) taking care of the adjustment in the control strategy.

  It is logged in the circular file “LOGIN.” of the Log File System

- MESSAGE 111: SYNCHRONISATION MESSAGE (for ADAPTIVE TRAFFIC SIGNALS)

  This message is sent out to the micro-simulator NEMIS to start a new simulation period.

  Every simulation period lasts 3 seconds (1 STEP) and the next simulation period starts when a new message 111 is received from the TCP MANAGER.

  It is logged in the circular file “LOGIN.” of the Log File System

All the messages in UTOPIA format are preceded by a 4 byte header that contains the following information:

Message Code	1 byte
Origin Micro	1 byte
Destination Micro	1 byte
Step/Cost	1 byte

Many run-time messages contain, after the header of the message itself, the send time of the message (using 3 bytes). This time is expressed in seconds from midnight.

If there are no different indications (message 93) the time bytes are ordered as follows:

LSB - MSB - MSB.

Figure 19 summarises the above information.

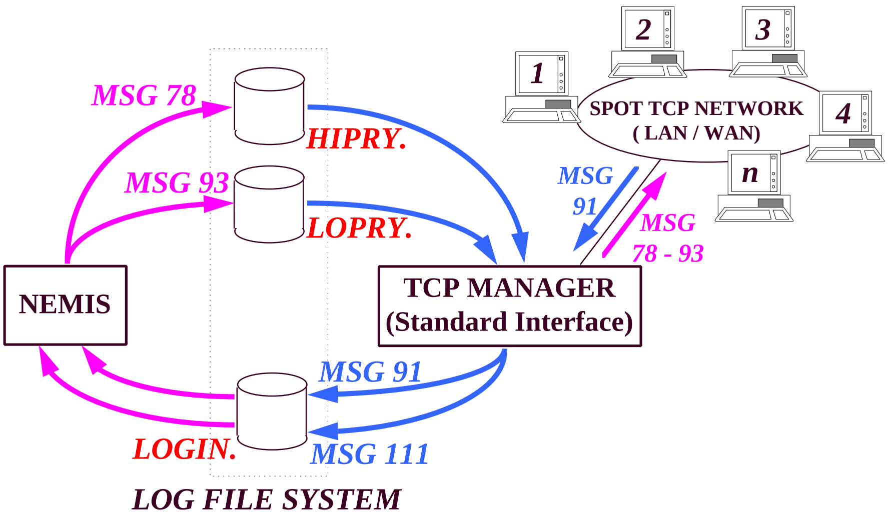


Figure 19: Description of the Log File System and exchanged Messages

- The Log File System Management Library

The ***L**og **F**ile **S**ystem **LIB**rary* manages the Log File System files. The system date and time variation are also managed by the library in order to avoid corruption of the Log files.

- The Simulation Process

The flow chart in Figure 20 (the same as in the introduction), shows in schematic way, the behaviour of the two main tasks of the TCP MANAGER and the interaction between NEMIS, the TCP MANAGER and the whole simulation network.

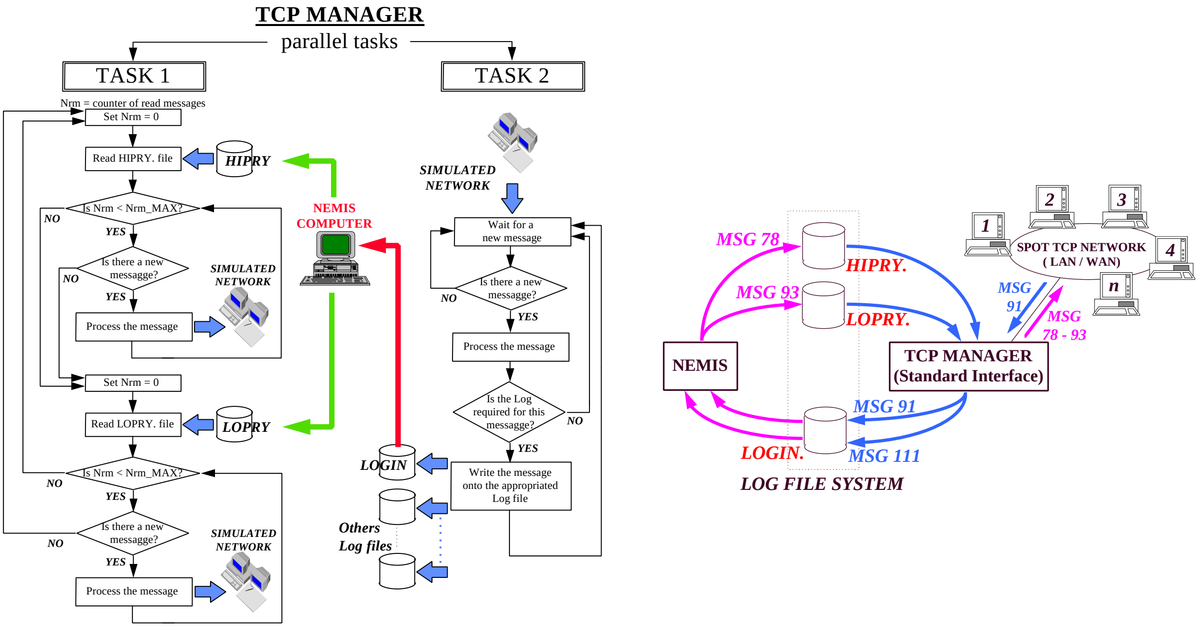
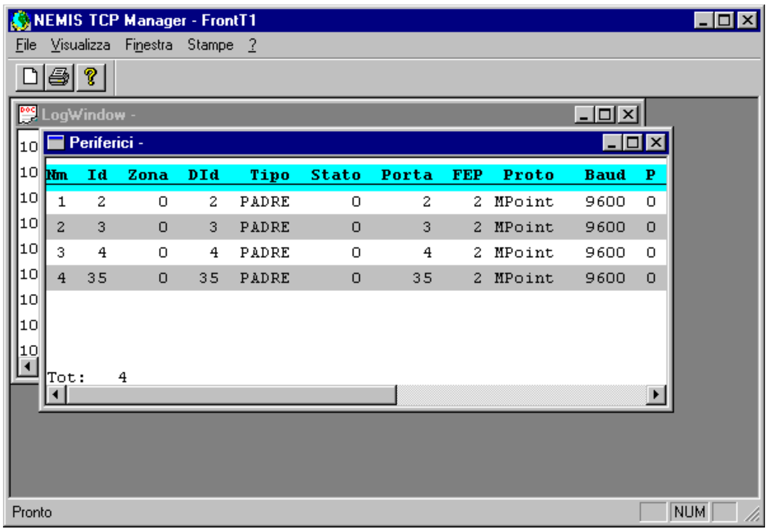


Figure 20: TCP MANAGER behaviour, Log File System and exchanged Messages

The behaviour of the communication between the TCP MANAGER and NEMIS, can be summarised as follows:

- The TCP MANAGER receives from the external control functions that reside within SPOT units or within other user-developed packages, all the messages containing the elaborated control strategy (Message 91). The communications between the TCP MANAGER and the external control functions are based on TCP/IP standard protocol.
- When the TCP MANAGER receives a new message, the message itself is processed and then written into the appropriate file in the Log File System.
- The TCP MANAGER also reads the circular files HIPRY and LOPRY where NEMIS writes the messages needed by the external control functions (Messages 78 and 93).
- When a new message is written by NEMIS into a command file of the Log File System, the TCP MANAGER, processes the message and then sends it out towards the appropriate SPOT unit or to the appropriate external control function.

Figure 21 shows a screen-shot of the TCP Manager user interface.



Nm	Id	Zona	DId	Tipo	Stato	Porta	FEP	Proto	Baud	P
1	2	0	2	PADRE	0	2	2	MPoint	9600	0
2	3	0	3	PADRE	0	3	2	MPoint	9600	0
3	4	0	4	PADRE	0	4	2	MPoint	9600	0
4	35	0	35	PADRE	0	35	2	MPoint	9600	0

Tot: 4

Figure 21: User interface of the NEMIS TCP Manager

The way in which the simulation process changes when NEMIS is linked with external control strategies can be described as follows:

1. NEMIS simulates for the three seconds and then:

    prepares the messages 93 (traffic counts for external adaptive control strategy)

    prepares the messages 78 (PT vehicle forecasts for external PT priority strategy)

2. NEMIS writes onto the Log File System the messages previously prepared

    all the messages 93 are written onto “LOPRY.” file

    all the messages 78 are written onto “HIPRY.” file

3. At the end of the simulation period, NEMIS reads the circular file “LOGIN.” starting from the last message read and looking for messages 91 coming from the SPOT local units and for message 111 coming from the TCP Manager.
4. When a message 91 is detected the SEM6 matrix (containing the traffic signal information) is updated with the new planned plan information.
5. When a message 111 is found, the new simulation period (that lasts three seconds) starts and the simulation process returns to the step 2.

Figure 22 shows a flow chart for the simulation process.

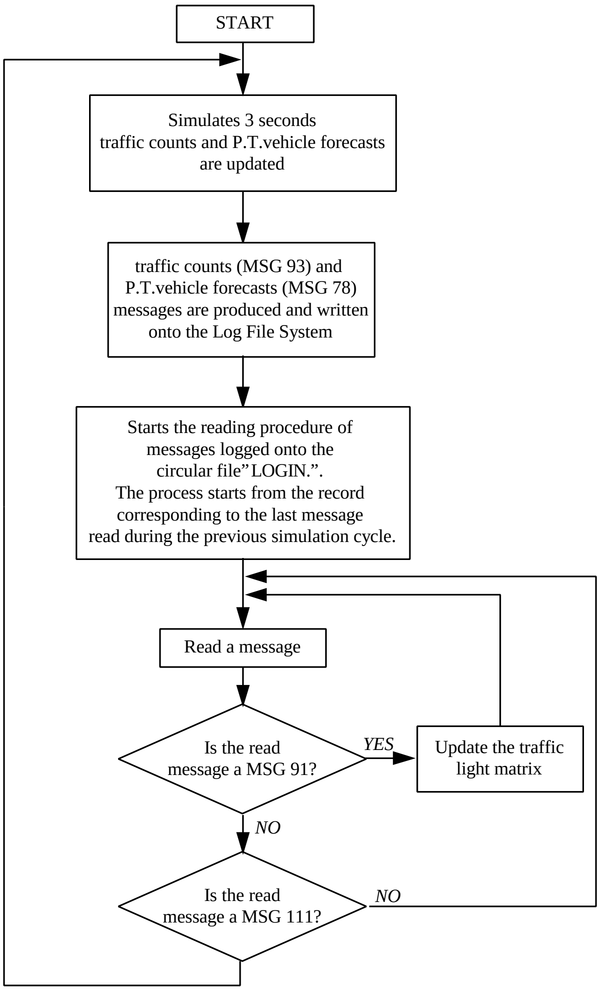


Figure 22: NEMIS simulation process

All the messages exchanged between NEMIS and the TCP MANAGER are logged in the Log File System. Therefore together with the standard output provided by NEMIS for private traffic and public transport analysis (See NEMIS Manual), it is possible to use all the UTOPIA analysis tools to evaluate the impact of the external control strategies on traffic mobility and on public transport priority. Figure 23 shows some examples.

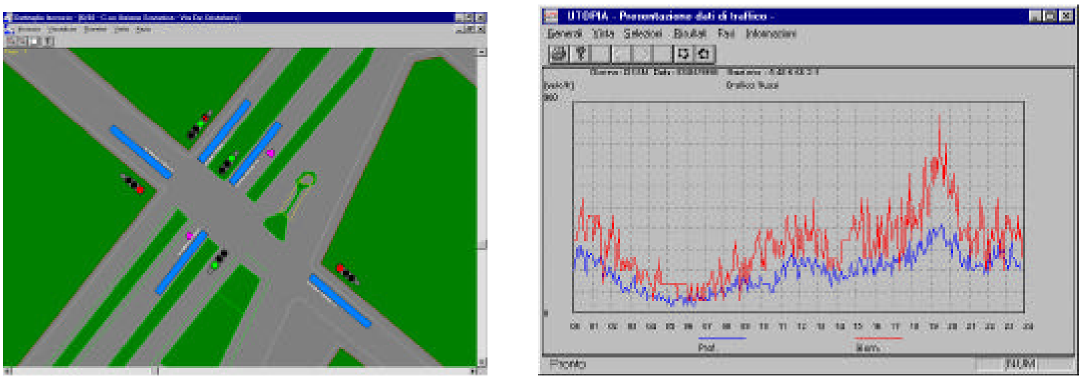


Figure 23: Example of UTOPIA analysis tools

Variable Message Signs

The main objective of the use of Variable Message Signs for traffic guidance is to support drivers by dynamic and collective information about suitable directions to reach their destinations.

Within the extensive concept of Collective Traffic Information, the purpose of variable message signs is to provide drivers with information of general interest concerning current and foreseen problems in the network - such as roadworks and limitations to traffic circulation. This information does not necessarily include suggestions about the route to follow.

Focusing on Collective Traffic Guidance applications, a significant influence on traffic behaviour is achieved by placing the signs at strategic points of the road network, in such a way as to intercept the main traffic flows. Then the Traffic Guidance applications provide additional information on the *causes of the diversions* when the directions suggested differ from those "normally" chosen by traffic on the basis of the network knowledge, or from those suggested by static signs.

Micro-simulation models performing the "verification" (operational tests and impact analyses) of guidance strategies should have the following characteristics:

• drivers behavioural model review	introduction of stochastic processes suitable for representing drivers compliance with panel information
• data structure definition	introduction of new data structure to model VMS panels and their information
• VMS loading and updating procedures	introduction of new procedures related to the new data structures loading and updating operations

- definition of the interaction between VMS panels and control function
- definition of data that need to be collected during the simulation to assess control strategy effects

NEMIS already supports a model for the simulation of VMS effects on traffic behaviour. In this case the model will be further calibrated focusing attention on the parameters characterising the control strategy operations.

It would seem from reading the previous sections that the VMS model is based on the aggregation of micro-destinations in macro-destinations. In fact, the destinations addressed by the VMS are selected in such a way to meet the interest of main traffic flows crossing the sites where the signs are placed. Therefore VMS control strategies must be able to model and elaborate traffic diversions towards destinations that in general correspond to groups of elements of the road network and that can be defined as "*macro-destinations*".

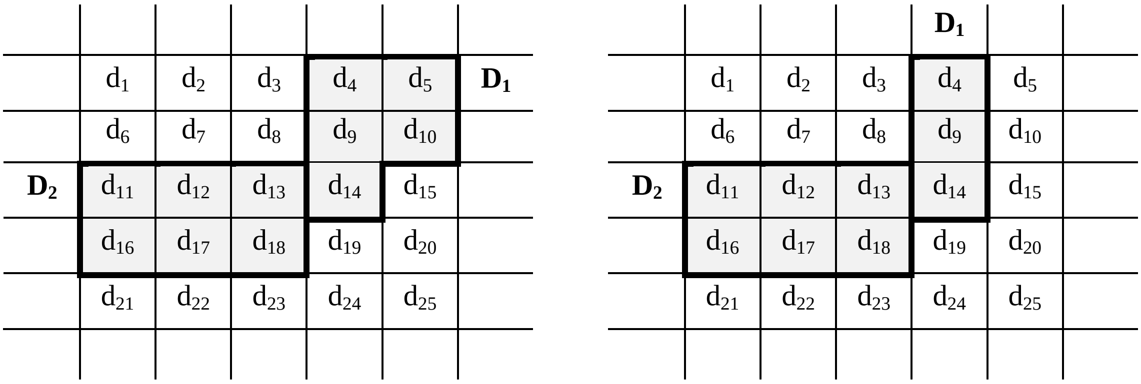
The destination of the driver corresponds to a particular point (or limited zone) of the network. For micro-simulation purposes this destination is modelled in terms of nodes. Consequently, driver destinations can be defined as "*micro-destinations*".

Therefore a correspondence between macro and micro destinations is defined. This is needed both to implement the model of the interaction between drivers and VMS (the driver needs to identify the possible macro-destination which corresponds or includes his micro-destination) and to fix the area addressable by the guidance strategy by means of each VMS.

Defined:

$\mathbf{d_i}$	the generic micro-destination
$\mathbf{D_j}$	the generic macro-destination

The following table shows an example of the correspondence between the macro and micro destinations of a VMS_1 and a VMS_2.



VMS_1 Macro-destination D_1 = aggregation of micro-destinations (d_4 , d_5 , d_9 , d_{10} , d_{14})

VMS_1 Macro-destination D_2 = aggregation of micro-destinations (d_{11} , d_{12} , d_{13} , d_{16} , d_{17} , d_{18})

VMS_2 Macro-destination D_1 = aggregation of micro-destinations (d_9 , d_{10} , d_{14})

VMS_2 Macro-destination D_2 = aggregation of micro-destinations (d_{11} , d_{12} , d_{13} , d_{16} , d_{17} , d_{18})

Table 16 : Macro-destinations (D) and micro-destinations (d)

Table 16 shows that the same macro destination can be addressed by different VMS, but due to the different VMS positions the common macro-destination could correspond to a different aggregation of micro-destinations.

The VMS control strategy can be subdivided into the following modules:

- control function
- actuation module
- driver behavioural model

The *control function* runs every 5 simulation minutes and performs the following actions:

- on the basis of the observation of the density of critical links in network the turning percentages a_{ijd} for each carriageway and for each destination in the network are evaluated, starting from their nominal value a^r_{ijd}
- for each VMS panel, the turning percentages a_{ijd} are modified to turning percentages a_{ijD} related to the macro-destinations, using the carriageway flows for macro-destinations F_{iD} that are obtained by grouping the carriageway flows for micro-destinations F_{id}
- the turning percentages a_{ijD} are used to evaluate the time of permanence of the message on the VMS panel
- for each panel, for each turn and for each macro-destination the indicators of the diverted flow I_{id} are evaluated

The *actuation module* operates every second and maintains the suggested turns on the VMS panel for the time evaluated by the control function

The *driver behavioural models* operates updating the status of the vehicle every simulation step (1 sec) on the basis of the following item:

- status of the traffic signal at the end of the link and/or right way precedence rules
- desired turn at the end of the link (defined on the basis of pre assignment results (BASSOT) that aims to minimise the travel time in the network)
- particular control strategies operating on each vehicle (i.e. route guidance)
- movements allowed on the link (depending by the position of the vehicle on the link)

- car following rules
- The modifications fn driver behavioural models necessary to simulate the presence of a VMS panel on the link, are taken into account by the simulator during the assignment of the turn at the end of the link for the examined vehicle.
- When a VMS panel is placed on a link, the desired turn at the end of the link will be determined by taking into account the information shown by the panel, as well as the equilibrium assignment.
- The assigned turn depends on the destination **d** of the vehicle, and the correspondence between this destination and the macro-destination **D** addressed by the panel
- The message shown will be accepted/rejected in a stochastic way but will adhere to a mean compliance rate (user defined)

Figure 24 shows the driver behavioural model adopted

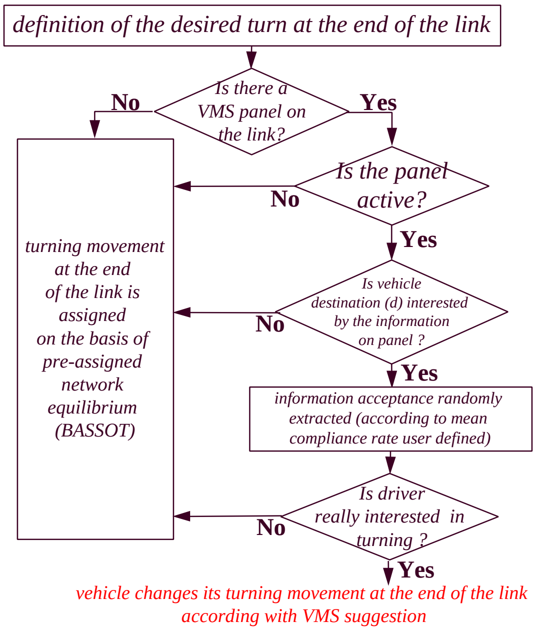


Figure 24: Driver behaviour model

For Collective Traffic Guidance Strategies, user compliance is very important. Compliance rates are being further validated by field trials on the basis of two different methods:

- indirect method:

Compliance rates are computed on the basis of re-routed traffic flows.

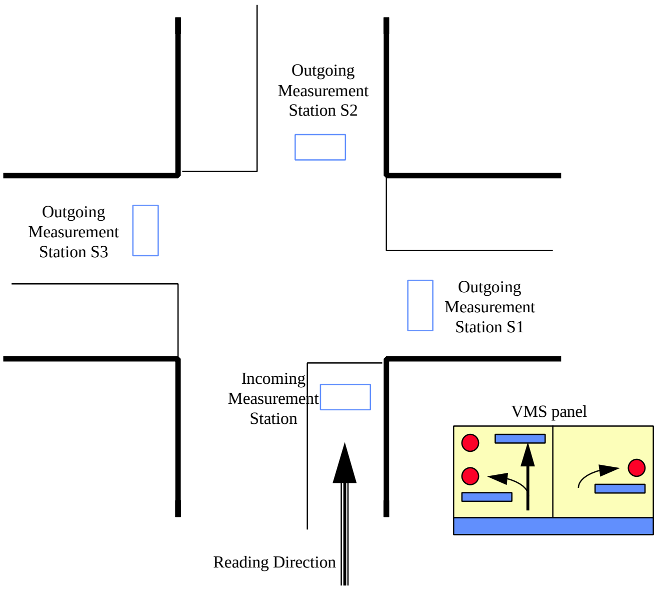
An ON/OFF approach can be used to gather traffic data in the two different operative conditions: system in operation (ON) and system not in operation (OFF).

During OFF measurements, the VMS must not be visible.

Carriageways downstream of the link where the VMS is located must be equipped with traffic detectors in order to gather traffic counts in both ON and OFF conditions (see the following scheme). These values will be used to compute turning percentages.

- direct method:

Compliance rates are computed by direct interview of drivers downstream of the VMS.



Dynamic Route Guidance

There is an interest in Individual Route Guidance systems simulation because of the general opinion that this kind of system will soon become an important instrument for Traffic Management.

There is interest in both in the possible impact of different systems features, architecture and penetration rates, and in the feasibility of integration of schemes involving IRG, UTC, VMS and other traffic and transport control systems.

Individual guidance information is provided to the driver by means of acoustic, optical or combined technologies. The best solution has not been fixed yet and depends both on the type of information to be communicated and on safety issues.

Individual guidance is provided according to static route definitions or dynamic route calculation. The dynamic solution is performed based on current and foreseen traffic conditions and is more related to Traffic Management concepts.

On the basis of the Individual Route Guidance systems (in the following simply referred to as IRG) classification provided in Deliverable 4 - Annex A Sec 9, NEMIS can simulate dynamic[4] autonomous[5] and infrastructured[6] systems. Also dual-mode[7] systems can be simulated.

---

[4] Travel time, traffic density and congestion are the parameters dynamically updated in the context of the dynamic IRG system.

[5] Decisions are taken according to the current vehicle position referred to the digitised territorial map, following pre-defined routes or taking into account possible dynamic traffic information (congestion, incidents, flows or travel times according to the system) provided by broadcasting systems.

[6] The system operates based on two-way communication between on-board equipment and roadside infrastructure (such as infrared beacons) connected to a centre. The ultimate "route choice" is performed on-board the vehicle according to the driver destination, while the "route calculation" is performed at the central level where dynamic traffic data are processed to update the network status estimate and to consequently optimise the routes for the possible O/D pairs.

[7] Dual mode is a combination of autonomous and infrastructured systems. The vehicle is able to perform the route choice on-board, based on the local database and on the traffic information transmitted by the broadcasting systems. When it crosses the area of the road side equipment it exchanges data and performs as in the case of the infrastructured solution.

All the simulations of infrastructured systems are based on Short Range Communication systems: communication performed by means of Infrared or Radio Beacons.

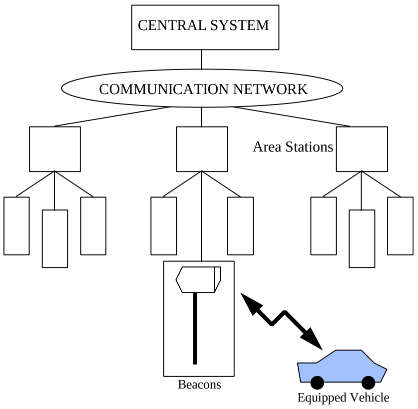


Figure 25: Infrastructured Route Guidance general architecture

In the adopted architecture, the global map is located at the infrastructure level. Small streets are not modelled. Positioning is performed on-board using autonomous equipment (dead reckoning and map matching functions), and dynamically via beacons. Traffic data (from vehicles and other sources) are centralised and refreshed with a sample period of few minutes. They are used together with historical data to compute the optimal routes.

Mono-routing and multi-routing criteria are used to define routes. In the mono-routing concept only one route is suggested to all the equipped vehicles going to the same destination. In the multi-routing concept the flow is split into several paths according to the possible (significant) alternatives.

Individual route guidance modelling involves the following aspects:

- the development of the module which performs the strategy for route calculation according to the optimal criteria adopted
- the development of the module which performs the route choice for the single vehicle
- the representation of the communication infrastructures (if any) which are located in the network
- the extension of the driver behaviour model to include the interaction with the on-board equipment
- the development of the data filtering module that acts as the interface between the network/traffic model and the guidance strategy module.
- the development of the scheduler which defines the timings of exchange of information (if any) between vehicles and infrastructure
- the extension of the traffic model to include the new typology of equipped-vehicles, the related generation procedure and the connection with route choice activities.

The main part of these models, already supported by NEMIS, have been revised and calibrated.

It must be underlined that the results in this field are only expected from the Dynamic IRG solutions, i.e. those normally referred to as DRG.

First of all we introduce the concept of a routing vehicle as used by NEMIS: a routing vehicle has the capability to elaborate information and take decisions. A routing vehicle knows (as does a normal one) its final destination and, while no information is received from external control strategies, it behaves as a normal vehicle, trying to achieve its final destination. As soon as IRG information is received from an infrastructure of the network (such as IRED beacon), the routing vehicle calculates the best route for its destination. While no further information is received, the routing vehicle follows the best route for its destination calculated during the last elaboration. It follows that the route taken by a routing vehicle depends on the elaboration of all the available information and so, due to the fact that information is provided by the external control strategy, on the routing strategy adopted.

Roadside Infrastructure Model, Driver/Vehicle model, RG Vehicle generation and the management of the RG data flows are integrated within the network/traffic model due to their direct correspondence with models that are already important components in the micro-simulation model.

NEMIS supports the following DRG control strategies:

- *Multi Path Fully equipped Network.* Each time that a routing vehicle approaches an intersection, it receives the information regarding the next turning movement (based on its destination). This approach can be compared to the assumption of a fully equipped network, where each intersection is equipped with a beacon.
- *Multi-Path Algorithm (MPA).* It is assumed that only some intersections within the network are equipped with an IRED beacon, so that a routing vehicle can receive the information needed to choose the route (desired turning percentages) only when it is approaching an equipped intersection. The effective route choice is performed on-board. This approach (which is more realistic than the previous one) supposes the presence of a communication system able to manage the exchange of a great amount of data between routing vehicles and beacons.
- *Time sharing Mono Routing.* Together with the assumption that only some intersections within the network are equipped with an IRED beacon, here it is assumed that only one path is suggested to each routing vehicle. The choice of the route to be suggested is performed on a time period greater than the time needed to perform the evaluation of turning percentages. For each route, starting from the desired turning percentages, an attribution percentage is evaluated (β); then, this attribution percentage is converted into the time period during which the corresponding route is suggested to the vehicles. It follows that, at any time, only one route is suggested to all the routing vehicles that have the same desired destination; different suggestions can be provided at different moments.
- *Mono Routing "max-beta".* Similar to the preceding solution for the route calculation method, it is different because during all the time period needed to perform the evaluation of turning percentages only one route (those maximising the attribution percentage β) is suggested to all the routing vehicles that have the same destination.
- *Minimum Time.* This algorithm is not properly a DRG approach. It is based on the evaluation of the shortest path between beacons and all reachable destinations. The minimum path is then communicated to all equipped vehicles crossing the intersection where beacons are located. It is assumed that the current travel time on each link[8]as well as incidents and congestion phenomena are known.
- *Dual-Mode Route Guidance.* As the preceding solution, this last algorithm is based on minimum path evaluation. It simulates the behaviour of two different strategies operating at the same time within the simulated network. *RDS/TMC* technology transmits to the whole network, information related to local congestion (in terms of link impedance), with 20 minutes of delay. *Beacons* transmit minimum paths to reach all destinations, evaluated on the basis of the current network status, with 10 minutes of delay. Equipped vehicles normally follow the minimum path autonomously evaluated on the basis of nominal travel time and of all information coming from the RDS/TMC system; then, when they cross an intersection where a beacon is placed, they receive all the information on optimal path to reach their destination.

The control strategy applied is common to all the above solutions, and it is based on the calibration of the link density to a nominal value. Also the calculation of the impedance and cost functions, as the calculations of the desired turning percentages is common to all solutions.

---

[8] In NEMIS this times are evaluated on the basis of the time employed to travel the link by preceding vehicles

Figure 26 shows in a schematic way the behaviour of DRG strategies

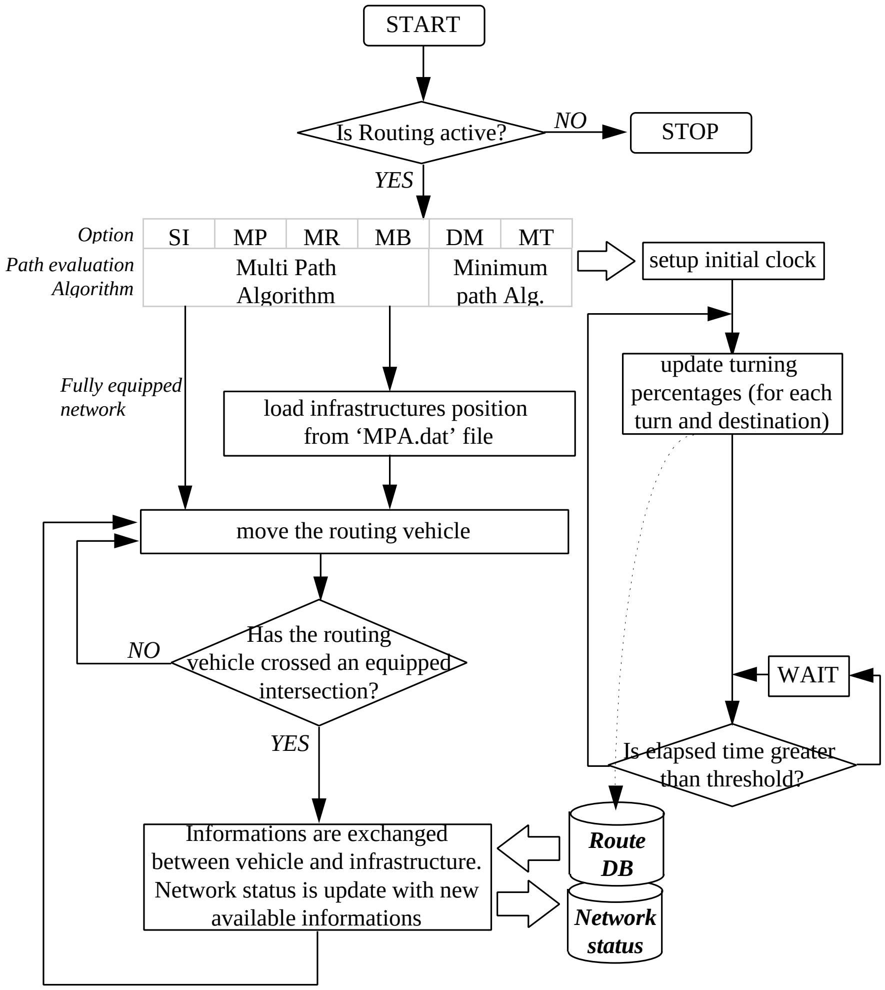


Figure 26: DRG schematic behaviour

SITRA-B+

Introduction

In accordance with the Update Specifications proposed in Project Deliverable D4, the following functions were developed or enhanced in SITRA-B+ :

- *Public Transport Services*
- *Roundabout*
- *Parking Management*
- *Adaptive Traffic Signals*
- *Public Transport Priority*
- *Variable Message Signs*
- *Incident Management*

The improvements to the *Public Transport Services* modelling consist of a better definition of routes, schedules and stops. Bus routes are described as a fixed series of links from an origin to a destination. Schedule definition is frequency-based with possible random deviations. A new type of bus stop is created: the bus stop lay-by, including new behaviour rules for pulling into or out at the bus stop.

A complete *Roundabout* model was implemented in SITRA-B+. This new development addresses both driver and vehicle behaviour models and graphical user interface functions. Simple rules were defined in order to deal with lane changing decisions both approaching and driving in the roundabout, and new behaviour parameters were introduced for the gap acceptance model. Video data from a test site in Toulouse has been used for the validation of this roundabout model.

The *Parking Management* model improvements mainly deal with street parking management. Street parking (along the roadside) is no longer modelled by destination or origin nodes, but as intermediary destination nodes with a given stopping probability. A series of parking spaces at precise locations is attached to each street parking node (which is itself attached to a given lane). Mean and standard deviations of parking duration are parameters that can be selected by the user for each street parking set.

The *Adaptive Traffic Signals* improvements consist in the implementation of the new traffic signals description and management protocols presented in Deliverable D4 « Update Specifications ». It thus increases the range of UTC strategies able to be linked with SITRA-B+, such as the possibility to alternatively run fixed time plans and to interrupt them by adaptive sequences.

The development of new specialised detectors dedicated to public transport vehicles now allows us to consider a wider set of *Public Transport Priority* strategies that can be tested with the microscopic traffic simulator. Formatted messages are generated and stored in data files ready to be used by the external PT priority strategies.

As far as *Variable Message Signs* are concerned, a new class has been created for VMS modelling, and dynamic route guidance purposes were associated to this new object. Guidance controls to be displayed on the VMS are calculated and sent to SITRA-B+ by an external strategy, together with modified routes and compliance rate for concerned vehicles.

Finally, concerning *Incident Management* features, the possibility to generate scheduled incidents (location, occurrence time, duration) was added in SITRA-B+. This new feature is particularly well suited for testing the robustness of UTC strategies and the ability to react to unpredictable events.

Public Transport Services

In the former version of SITRA-B+, it was possible to associate only one bus stop with a given bus route, and vehicles were generated according to a deterministic period (without random variations).

Stopping time was also constant, and only typical bus stops were modelled, thus causing systematic blockage for following traffic.

The new developments provide a more complete and more realistic description of *Public Transport Services*, both for route schedule, bus stop layout and pulling out behaviour.

Route schedules are still given by starting and ending time and a theoretical frequency, but a random parameter (standard deviation of the time period) is added in order to model the usual irregularities. The bus generation module uses these parameters. A null value for the time period standard deviation would mean that the generation node is a terminus.

There is no longer any limitation on the number of bus stops per route. Each of them is attached to a given link and to a given route, and other parameters are the position on the link, the mean and standard deviation of stop time, and the layout parameter (*typical* or *lay-by*).

In the case of a bus stop lay-by, a pulling into and a pulling out behaviour model were implemented. The pulling out model allows the following traffic to stop for a few seconds before the end of the bus stop time, and the pulling out manoeuvre takes place as soon as the lane is cleared along the bus stop location. The animated graphical display enables the proper behaviour of the model to be checked.

A set of outputs is available at the end of a run, which can be used for example to analyse the effect of a given UTC strategy on the journey time and regularity.

Three main functions were developed in order to improve *Public Transport Services* management in SITRA-B+: vehicle departure and stop time generation, pulling into model and pulling out model for bus stops lay-by.

- Vehicle departure and stop time generation

For each PT route, a table containing the future departure times is generated during the initialisation phase. Each theoretical departure time is altered with a truncated Gaussian noise value, whose standard deviation value is given in file *vehicle_schedule.rel*.

A similar procedure is applied for stop times : each time a new bus is generated at an input node of the network, a list of stops is created, including the stop time value which is calculated as a truncated Gaussian value taking into account the mean and standard deviation values given in the input file. As different values can be assigned to each bus stop, this allows the effect of disturbances generated by different levels of passenger demand to be evaluated.

- Pulling into algorithm (bus stop lay-by)

When a bus reaches the stop position (the stop is supposed to be « reached » when the distance between the front of the bus and the stop position is less than a given threshold), it becomes « transparent » for the following traffic (case of a bus stop lay-by), which means that it is no longer considered as the preceding vehicle by the following car. This procedure, which avoids adding supplementary lanes in the network description, offers a modelling capacity almost as complete as the one obtained with an explicit bus stop layout description. The graphical display of SITRA-B+ was modified in order to visualise the bus stop lay-by configuration (see next paragraph).

- Pulling out algorithm (bus stop lay-by)

In order to warn upstream traffic before the pulling out manoeuvre, and so initiating the creation of a gap, the following procedure is implemented : at a given number of seconds before the end of the scheduled stop time (fixed parameter of the model, not to be changed by the user), the bus looses its « transparency » : incoming traffic therefore decreases speed, which naturally leads to a gap creation in fluid conditions. Then, when the stop time has elapsed, bus leaves its stop after having checked the presence of an acceptable gap on its lane. If there is no acceptable gap (case of congested traffic situation), it waits until the queue is cleared.

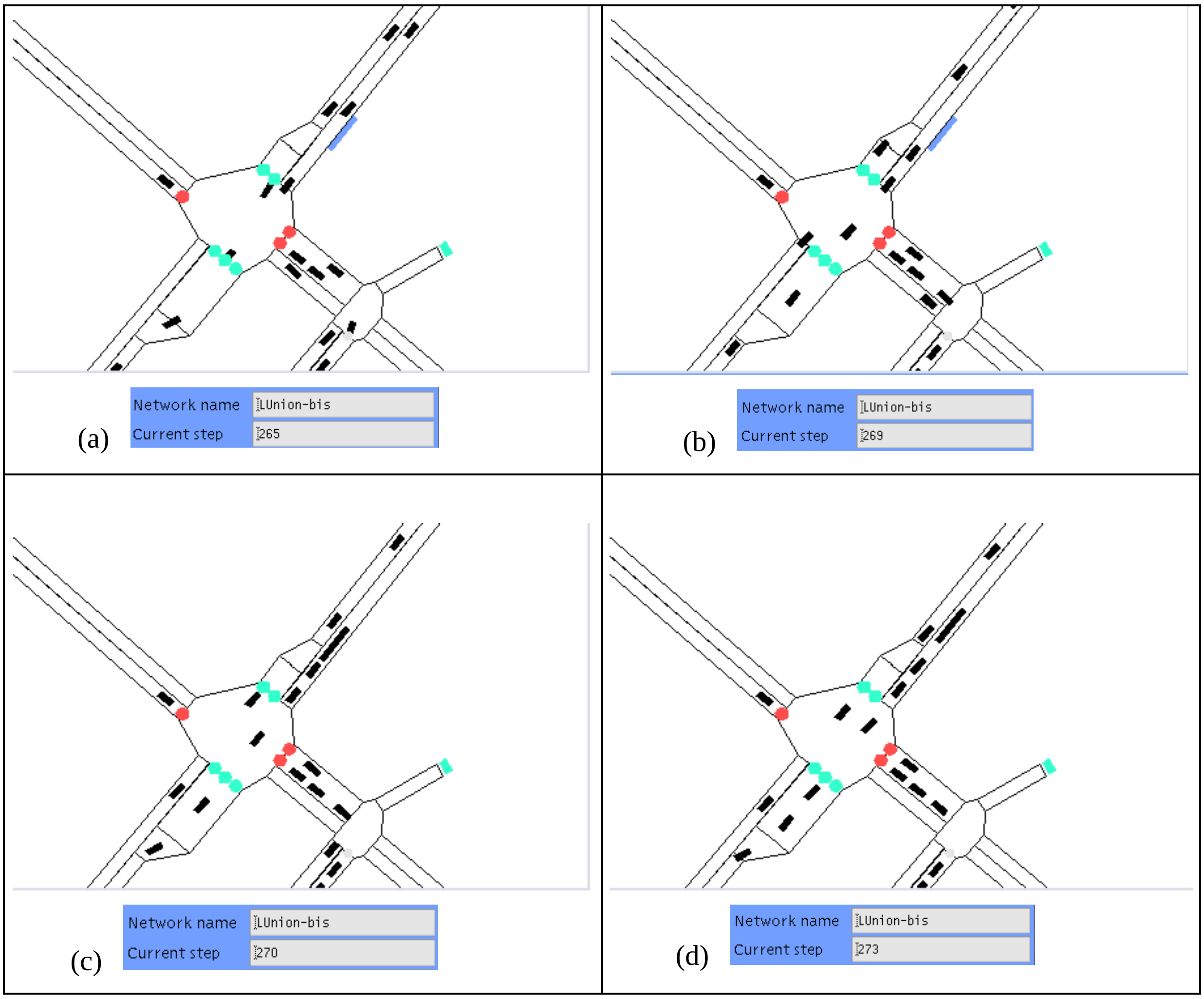
A sequence of screen shots from the animated graphical display of SITRA-B+ (UNIX Version) is shown in Figure 27, illustrating the various steps of a bus departure from a bus stop lay-by :

(a) : bus stopped ; incoming traffic running freely

(b) : incoming traffic being warned of bus departure

(c) : bus leaving the stop

(d) : a few seconds after bus departure


(a)

Network name LUnion-bis
Current step 265

(b)

Network name LUnion-bis
Current step 269

(c)

Network name LUnion-bis
Current step 270

(d)

Network name LUnion-bis
Current step 273

Figure 27 : The bus layby in operation

Data available from Toulouse area have been used to validate the model. Validation data includes average journey time and bus stop data such as mean and standard deviation of time period between buses.

Roundabout

The roundabout simulation model implemented in SITRA-B+ addresses « classical » or « conventional » roundabouts, as they are described in paragraph 5.2 of Deliverable D4 « Update Specifications ».

The topological description of roundabouts in SITRA-B+ uses the existing basic network description structures (links, intersections, link- and intersection-lanes), with a specific development related to the animated graphical display, which now allows curved links to be represented. New data fields were introduced to distinguish between new link categories or shape and priority rules.

Three new behaviour models were introduced in order to take into account driver behaviour at different levels : in the approaching phase (lane choice and gap acceptance models), and inside the roundabout (lane changing model). In order for the gap acceptance model to work, a new stochastic parameter was added to the ones associated with each vehicle : the *aggressiveness* parameter. The way it is used by this model is explained below.

Special attention has been made to the validation of the elementary models. This has been done by exploiting video recordings of a roundabout located close to the CERT offices in Toulouse.

The SITRA-B+ microscopic description of a network is based on the use of *lanes* connected by *connection points*. The priority rule to be used when changing lane (from a link lane to an intersection lane or vice-versa, not in the case of lateral lane changing) is thus given by the *connection point* nature. A new type of connection point was introduced: the LEFT_PRIORITY one.

Figure 28 shows an example of a roundabout layout. It includes three roundabout entrance links, 4 roundabout links, 3 outputs (« ordinary » links) and 4 intersections with their associated intersection lanes. Red dots show the location of LEFT_PRIORITY connection points.

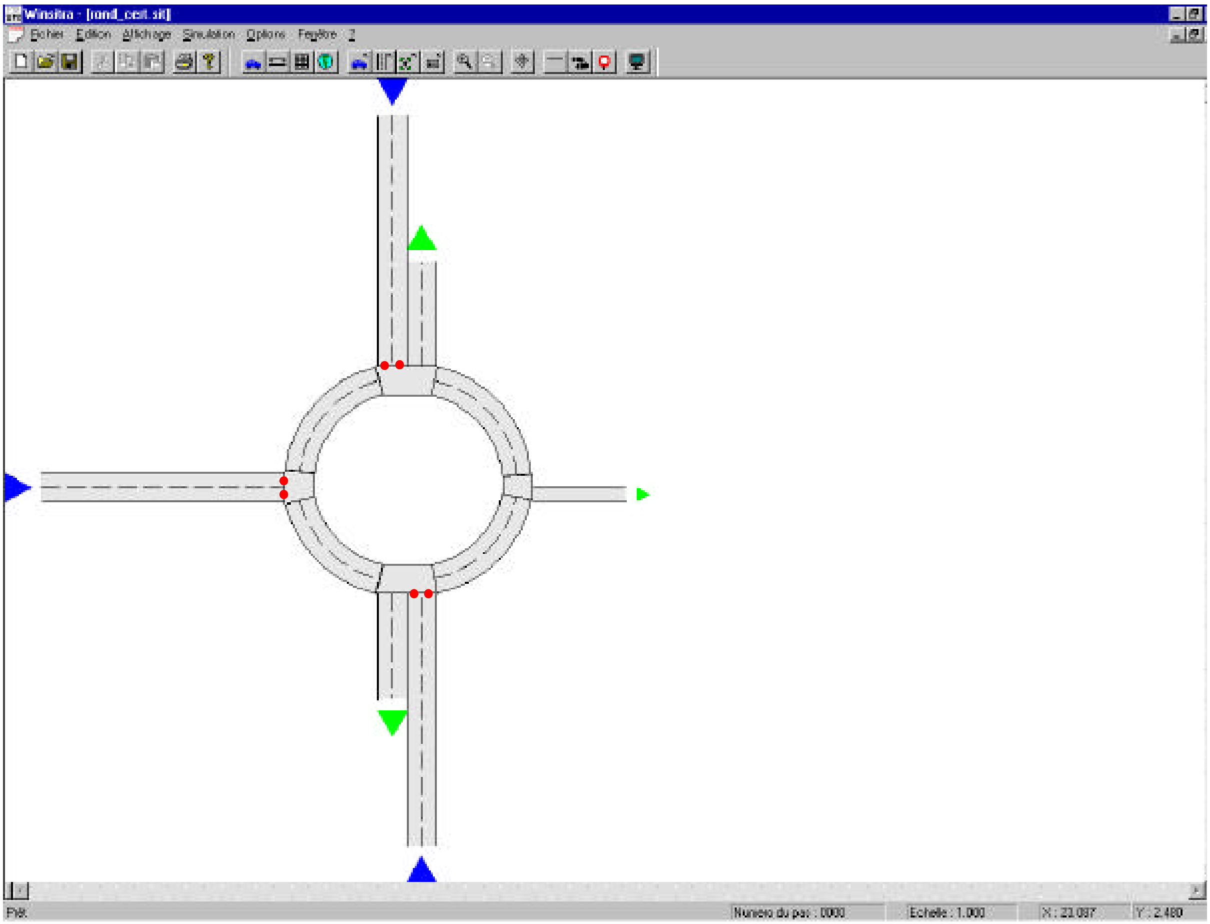


Figure 28 : Roundabout layout

The user of the SITRA-B+ simulation tool can adjust two sets of parameters so that the simulated behaviour matches the observed one.

The first one concerns aggressiveness modelling, which is a new individual stochastic parameter attached to each generated car in the network. The proposed scale goes from 0 to 10, and the user can specify its mean and standard deviation values by vehicle category. This parameter is then used to derive other attributes of the gap acceptance model.

The second set of parameters allows the user to choose the maximum and minimum values of the gap acceptance time (see also next paragraph for a more precise definition). These values are also given by vehicle category, and default values will be proposed to the user.

The new input data are thus:

- mean value of aggressiveness (0 to 10)
- standard deviation value of aggressiveness
- minimum gap acceptance time
- maximum gap acceptance time

New procedures have been implemented in SITRA-B+ to model the behaviour of drivers approaching and driving inside roundabouts. They have been added to the model in such a way that they do not interfere with existing procedures, which for example apply conflict rules inside intersections.

- Lane choice model

The first behaviour model is the lane choice model for vehicles entering the roundabout. The choice depends on the position of the exiting link, which implies that the vehicle destination and route have to be known. The following algorithm is implemented for each vehicle entering a new link :

```
If the vehicle is entering a RDB_ENTRANCE type link
    If the vehicle route follows only one ROUNDABOUT type link
        If the vehicle micro-route follows the farside lanes of those two links
            Compute a new micro-route for the vehicle, following the nearside lanes
        Endif
    Endif
    If the vehicle route follows more than two ROUNDABOUT type links
        If the vehicle micro-route follows the nearside lanes of the RDB_ENTRANCE type link and of the first ROUNDABOUT type link
            Compute a new micro-route for the vehicle, following the farside lanes
        Endif
    Endif
Endif
```

This approach assumes simple roundabout layouts, typically four-link roundabouts and two-lane links. The results gained from this model allow later consideration of more complex layouts.

- Gap acceptance model

Figure 29 illustrates the general principle of the gap acceptance model.

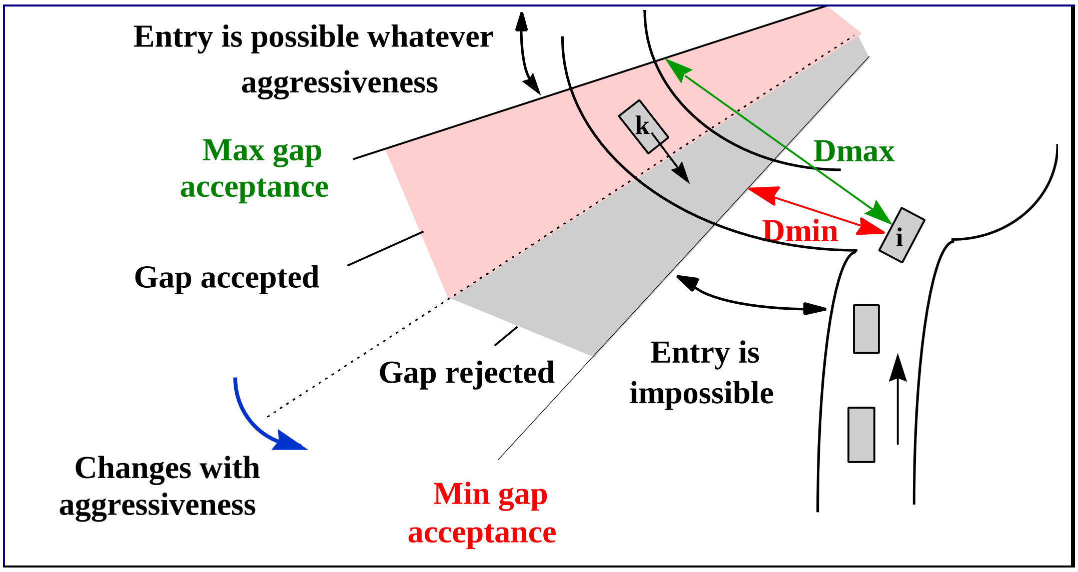


Figure 29 : Roundabout gap acceptance model

For a better understanding, the gap areas have been represented as distances and geometrical areas. They must however be transposed into time co-ordinates, where the previously defined parameters; *minimum gap acceptance time* and *maximum gap acceptance time* would be the transposed values of Dmin and Dmax in Figure 29. These distances are of course strongly dependent of the speed of approaching vehicle **k**.

The dotted line shows the initial value of the gap requested by entering vehicle **i.** This value is derived from the aggressiveness parameter attached to this vehicle when it was generated in the network. If vehicle **i** does not succeed in entering the roundabout with this initial gap acceptance value, and thus spends time queuing at the roundabout entrance, this value is reduced at a fixed rate until the minimum gap acceptance value is reached.

For each vehicle approaching a roundabout, a « decision distance » is calculated, from where it has to decide whether or not to enter the roundabout, and so to apply the gap acceptance model. This distance corresponds to the stopping distance of the approaching vehicle.

If the vehicle is not authorised to enter the roundabout, a « virtual stop » is generated at the LEFT_PRIORITY connection point located at the roundabout entrance. Then, at each time step, this vehicle keeps checking the gaps until entrance is possible.

- Lane changing model (inside the roundabout)

This model mainly concerns the lane changing from the farside to the nearside lane, for vehicles that have to drive more than a half circle. The proposed algorithm simply detects when the vehicle enters the last ROUNDABOUT link of its route, and computes a new micro-route following the nearside lane, as explained previously in the lane choice model. Nevertheless, if there is a path towards the desired exit using the farside lane and if the nearside lane is congested, the vehicle can keep to the farside lane until the exit.

The animated graphical display of SITRA-B+ allows the simulated behaviour of vehicles approaching and driving in the roundabout to be checked, and a comparison to be made with the real behaviour, using for example video recordings of a roundabout.

The chosen area is a Grade-Separated Interchange with One Bridge and two Roundabouts (see the terminology adopted in model update specifications for roundabout). One of these roundabouts has 3 entries (each with two lanes) and 3 exits (2 with two lanes and 1 with one lane). Furthermore it has 1 segregated lane that allows a part of traffic to go from an entry to the first exit. This roundabout is represented in Figure 30.

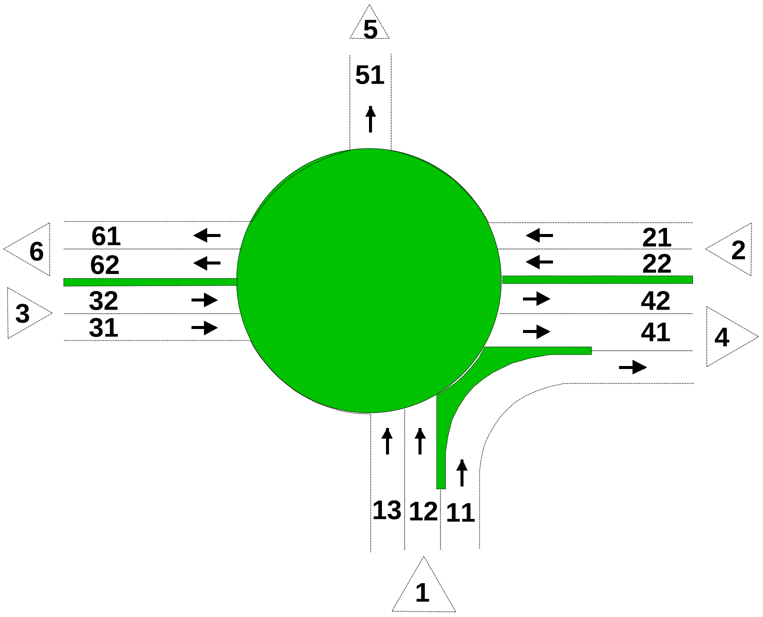


Figure 30 : The test roundabout

Entries and exits are represented with triangles. Entry and exit numbers are represented in each associated triangle. Lane direction is represented with a black arrow. Lane number is indicated on each lane.

Data collection has been performed by video from a point where all entries/exits are visible and with a video recording. Video data analysis has determined :

- traffic flow for each entry/exit
- for each Origin/Destination pair
  - traffic flow
  - average travel time
  - lane choice at roundabout entrance
  - lane changing inside the roundabout
- average travel time inside the roundabout
- average gap acceptance time for each entry
- driver behaviour near each entry

A sample of these data has been used to tune the roundabout model to represent real traffic conditions (lane choice, lane changing and gap acceptance). The remaining data has been used in simulation to check that the roundabout model developed in SITRA-B+ performs close to real conditions.

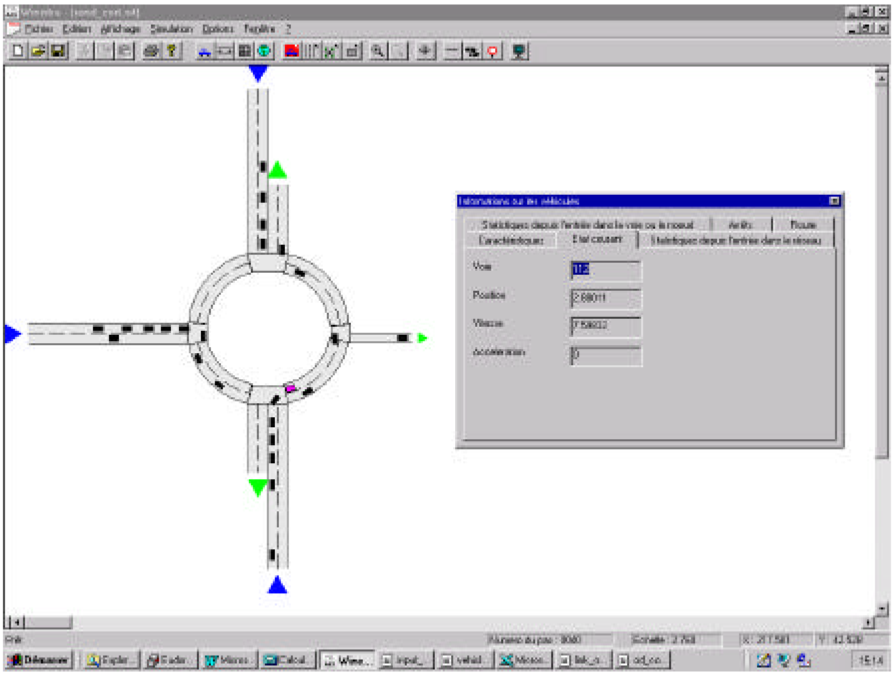


Figure 31 : The test roundabout in SITRA-B+

Table 17 shows the initial results obtained with the roundabout model of SITRA-B+, in order to check its correctness. In particular this testing relates to the gap acceptance model. Two different sets of values for minimum and maximum gap acceptance time were applied to the same demand on the test roundabout (5 minute time slice).

Those results show the significant sensitivity of the model to the gap acceptance time values, especially on the queuing time of vehicles approaching the roundabout on entry 2, without affecting the driving behaviour inside the roundabout (no significant difference on average speed and number of lane changing).

	Case 1	Case 2
min/max gap acceptance time (seconds)	1.0/3.0	2.0/5.0
mean value of observed gap acceptance times (entry 2)	1.77	3.14
standard deviation value of observed gap acceptance times (entry 2)	0.35	0.55
mean value of queuing time, in seconds (entry 2)	4.71	14.51
standard deviation value of queuing time, in seconds (entry 2)	9.87	31.80
number of vehicles having entered the roundabout	171	168
number of vehicles having exited the roundabout	164	163
average speed in the roundabout, in km/h	25.5	24.9
number of lane changing	42	43

Table 17 : How the gap acceptance time affects queueing time

Further data analysis gave the following results:

Traffic flow for each input (veh/h)

Input number	Simulated flow	Observed flow	Difference
1	1164	1296	10,2%
2	1068	1200	11,0%
3	468	612	23,5%
Total	2700	3108	13,1%

Traffic flow for each output (veh/h)

Output number	Simulated flow	Observed flow	Difference
4	216	240	10,0%
5	948	1440	34,2%
6	1080	1428	24,4%
Total	2244	3108	27,8%

Traffic flow for each O/D (veh/h)

O/D number	Simulated flow	Observed flow	Difference
1 - 5	384	576	33,33%
1 - 6	612	720	15,00%
2 - 5	372	516	27,91%
2 - 6	432	684	36,84%
3 - 4	216	240	10,00%
3 - 5	192	348	44,83%
3 - 6	24	24	0,00%

The results obtained from the simulation of this roundabout show differences in traffic flows from 0% to 44%. It is noticeable that the simulated flows at all entries are always less than the observed flows at the same points. This is due to the fact that the queues at entries were building (video is recorded at peak hour and queues are long at all entries) and the simulated roundabout is not able to let out as many vehicles as in real life. The "aggressiveness" and the "gap acceptance" parameters have to be adjusted in SITRA-B+ to consider the behaviour of drivers who are used to crossing this particular roundabout. The videotape observation also shows a more complex lane choice and lane changing behaviour than the simulated one, which contributes to limit the capacity.

However, travel times for vehicles crossing the roundabout, as well as the general behaviour of drivers within the roundabout were satisfactory.

Parking Management

This feature, which already existed in the preceding version of SITRA-B+ in a simplified form, has been enhanced to offer a more realistic representation of on-street parking. The car park type which is considered here is the « along the roadside » one (see Deliverable D4 page 55).

Two main improvements were made. The first one concerns the integration of the use of the street parking area by vehicles in relation with their trip inside the simulated network. Street parking is no longer considered as elementary origin or destination nodes, but as « intermediary » nodes in the vehicle route description. This allows a more precise description of the street parking itself. Parking spaces are clearly positioned along the street, and a procedure similar to the one developed for bus stop lay-by is used to model vehicle manoeuvres to occupy or free the parking space.

The user can assign mean and standard deviation values of parking duration to each street parking area, and the desired parking time of a given vehicle is a stochastic value drawn from the associated Gaussian law. The effective parking time can be greater during congested traffic situations. In order to be able to model various behaviours related to the parking manoeuvre, a « pulling in » duration can be specified by the user (a default value is proposed).

Two main steps can be distinguished in parking modelling : *getting in the car park* and *getting out of the car park.*

- Getting in the car park

Each time a vehicle enters a new link, the following algorithm is applied :

```
If the link is associated to a street parking node
    If this node is the next node in the list of parking nodes of the vehicle route
        Decide if vehicle will stop (using stopping probability attached to the node)
        If vehicle is going to stop
            Calculate (if necessary) a new micro-route using the nearside lane of
            the link
            If no free parking space is available
                Renounce to decision to park and continue the trip
            Else
                Choose randomly a parking space among the free spaces
                When the chosen space is reached, leave the vehicle stopped on
                 the lane during pulling in time value, then put it in the parking
                space
            Endif
        Endif
    Endif
Endif
```

The procedure used for parked vehicles is the same as that used for the bus stop lay-by (« transparency » indicator).

- Getting out of the car park

The same algorithm as the one used for buses to pull out from a bus stop lay-by is used. As soon as a space is freed, it is added to the list of free parking spaces for incoming vehicles.

Adaptive Traffic Signals

In accordance with the Update Specifications of Adaptive Traffic Signals presented in Deliverable D4 (see page 99 and following), the new developments made in SITRA-B+ concern traffic signal modelling and traffic controller description and operation.

Concerning the modelling part, a new data structure now has the « colour » class as the basic class for traffic signal description, and, if necessary, to it is possible to associate a dedicated behaviour to each colour. The derived classes then lead to the new controller and fixed time plan representations.

Concerning traffic controller operations, the implemented procedures enable the full exploitation of the new plan description (impulse based) and thus increase the range of strategies to be linked with SITRA-B+ (it is of course assumed that adaptive strategies are external entities).

An event-driven type approach is used to implement the new traffic signals management in SITRA-B+. Two types of events are associated with controllers and traffic signals : impulse occurring time and colour changing time. At each simulation step, intersection controllers are examined, and the following procedure is applied to each of them :

```
For each traffic signal do
    If time == next colour changing time
        Process colour changing
    Endif
Enddo
If time == next impulse occurring time then
    Calculate next impulse time (case where a fixed time plan is running)
    For each traffic signal do
        Process colour changing (if required)
        Calculate next colour changing times and next colour values
    Enddo
Endif
```

The advantage of this approach is that the same procedure is used both with fixed time plans and external adaptive strategies. The only difference concerns the next impulse time calculation, which holds only when a fixed time plan is running.

As there is no input data file to allow the direct initialisation of the traffic signal states, a start-up procedure was introduced to achieve this task, simply running a blank cycle before starting the simulation itself.

The traffic signal colour changes can be observed on the graphical user interface of SITRA-B+. The PC version also allows the current parameters associated with a traffic signal or with an intersection controller to be displayed.

Public Transport Priority

The main developments which were undertaken to improve *Public Transport Priority* management with SITRA-B+ are related to the modelling of Public Transport vehicle localisation procedures and to the implementation of a new detector type. As in the case of Adaptive Signals, the strategy producing or altering the traffic signal settings is considered as an external strategy, which can in this case receive new types of messages produced by the bus localisation procedure.

The choice of an « asynchronous » communication mode in file *modality.rel* for a given vehicle category means that the vehicle position will be known by the external strategy only when it reaches a LOC_BEACON type detector : there is thus no need of position calculation in this case.

When a « synchronous » communication mode is chosen, the vehicle is supposed to transmit its position at regular time intervals : functions *give_position_abs* or *give_position_rel* are thus activated, depending on the localisation system type :

- function *give_position_abs* : a random value, derived from the localisation model parameters given in file *modality.rel* is added to the true vehicle position
- function *give_position_rel* : a random value of the odometer drift is calculated using the corresponding parameters and attached to the vehicle when it is generated in the network. This value is then multiplied by the distance travelled by the vehicle since the last LOC_BEACON crossing, and added to the true vehicle position. LOC_BEACON are thus in this case « resetting » beacons for the travelled distance.

Variable Message Signs

The modelling capabilities of SITRA-B+ concerning dynamic information and guidance systems have been extended in order to be able to deal with Variable Message Signs. As for on-board route guidance systems, already supported by SITRA-B+, guidance messages are assumed to be generated by an external strategy, together with the new proposed routes and the estimated compliance rate.

All vehicles passing by the VMS location see and read the message, which displays a route guidance type message (concerned destination, advised route). VMS are modelled as a new kind of beacon, and a new function is added at the vehicle level in order to first identify the concerned vehicles (going towards the same destination but using a route different from the one advised), and then divert them, taking into account the obedience coefficient proposed by the external strategy.

VMS locations are displayed on the graphical user interface, together with associated messages. New output text files are introduced to check the effects of the collective route guidance strategies.

Each time a vehicle belonging to a category which is able to react to a VMS message passes by a VMS location, the following procedure is applied :

```
If the vehicle destination is the advised one
    If the initial vehicle route does not use the advised one
        Generate a random number between 0.0 and 1.0
        If this number is less or equal to compliance rate
            Assign the vehicle to the advised route
        Endif
    Endif
Endif
```

VMS locations are shown on the Graphical User Interface using special icons, and the current message can be displayed by clicking on it.

A scenario conducted on the Toulouse test site (see Deliverable 4, page 22), using 10 urban VMS, was used to check these new VMS dedicated functions. The strategy computes guidance recommendations (to turn right, left or to go straight on at the next intersection for a given destination) for all signs.

Incident Management

In SITRA-B+, the new developments related to *Incident Management* only deal with incident generation. Strictly speaking we do not address incident management strategies, but rather consider how UTC strategies can react to unpredictable events such as incidents.

Incidents are modelled by stopping vehicles at given times and at given locations, which remain stopped during a given duration.

Scenarios implying incidents are deterministic, which means that incident location, time of occurrence and duration are pre-defined by the user using a special input file.

At the beginning of the simulation, incident starting times are read from the appropriate input file and put in an ordered list of scheduled incident generation events.

At the beginning of each time step, current time is compared to the next scheduled incident generation time. If this time is reached, the first upstream vehicle driving on the concerned lane and able to stop at the incident location is looked for, and a "virtual stop" is attached to it; the vehicle is linked to the processed incident of the incident list and, at incident ending time, the "virtual stop" is reset.

BEST PRACTICE MANUAL

Introduction

A best practice manual of modelling and micro-simulation tools was written. The objective of the manual was to support users in dealing with specific traffic management problems such as congestion, shock waves caused by traffic disruption, harmful emissions etc. One group of problems is related to avoidance of secondary accidents and maintaining road capacity following incidents. Another group is related to network capacity. A third group of problems is related to warning and advice. Demand oriented problems such as trip planning and automatic debiting were not dealt with.

It is envisaged that the main users of the manual will be transport practitioners working in local authorities, central government and consultancies, as well as transport researchers and academics. The main users up to now seem to be researchers and only to a lesser extent local authorities. An objective for the Best Practice Manual is to enhance the use of micro-simulation for suitable dynamic transport problems and also to reach decision-makers as well as model users.

Simulation tools can be used on-line for dynamic traffic management or off-line for design and testing of control strategies. Chapter 3 of the manual discusses the steps required to perform a traffic simulation study. This includes the choice of the area to be modelled, the collection of data to calibrate and validate the model and the analysis of the model outputs. Specific user requirements that are discussed in Chapter 4, are user friendliness, short lead-time before use, validated results, limited need for expensive data acquisition and high cost effectiveness when comparing quality of result and resources in time and money spent in the simulation. Based on the review of existing micro-simulation models a set of guidelines for selecting a suitable micro-simulation model is presented in Chapter 5.

In the Stockholm test-site, results from macro simulation as well as micro simulation runs are available. This has given an opportunity to compare the two different approaches, which gives an idea of the improvement in accuracy that can be expected from micro-simulation modelling. This question is discussed in Chapter 6. Chapter 7 describes the data you need when you work with a micro simulation model and the different ways to collect this data. Calibration and validation guidelines are presented in Chapter 8. This includes examples from four of the SMARTEST test sites; Stockholm, Toulouse, Barcelona and Leeds. Both dynamic data and static data are needed for validation. Experience from the transferability study is also presented. A spread of European values for parameters, e.g. car following, desired speeds, emission rates etc. are provided. Further details on the processes involved in the formulation of scheme objectives are presented in Chapter 9. Examples of how the SMARTEST micro-simulation models have been enhanced to meet users' requirements are presented in Chapter 10.

Given the results from the evaluation and validation at the test sites and the comparison between macro and micro modelling, recommendations are made in Chapter 11. They concern when and how micro simulation models should be used in assessing the benefits of ITS investments as well as advanced control strategies for traffic management.

Transferability

Introduction

In SMARTEST, we had an ambition to look at transferability. If we have a model developed using data at one site, how can this help with the analysis at another site, if we use the same micro-simulation models to assess different schemes in other towns? How confident can we be that the conclusions we draw at one site are valid at another site? The new site might be in a region with a different set of objectives, so different indicators will need to be used. Two transferability tests were conducted. The first concerned the AIMSUN2 implementation in Stockholm, the second looked at the use of DRACULA, NEMIS and AIMSUN2 on a site in Leeds.

DRACULA, NEMIS and AIMSUN2 in Leeds

Introduction

Three micro-simulation tools have been used in the study, namely:

- DRACULA
- NEMIS
- AIMSUN2

All three tools were used to model a road network in Leeds. Comparisons were made between the outputs of each of the tools and data collected from the real road network.

Other issues addressed include:

- the ability of each of the tools to model all the features found in the test network,
- whether default values for calibration parameters such as those for car following are valid at the test site and
- the sensitivity and robustness of the results.

The test network

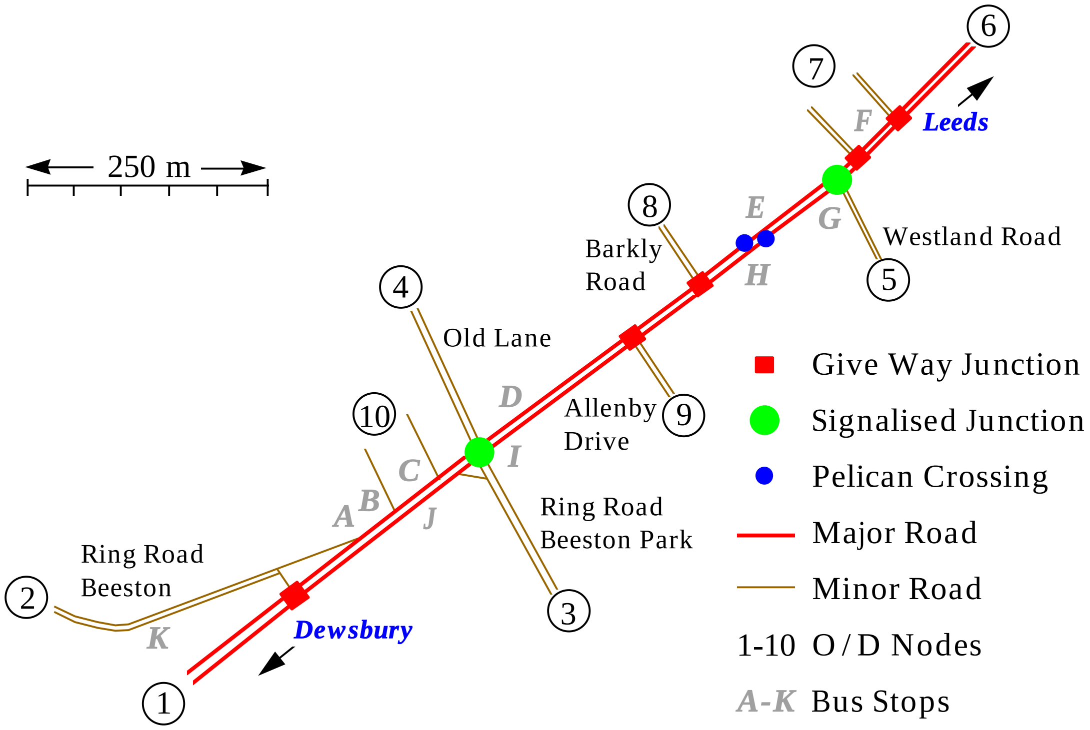


Figure 32: The Dewsbury Road - Leeds

A network in Leeds was chosen for this study because of the ready availability of suitable data to both define the network and to compare against model outputs. This data had been collected as part of the PRIMAVERA project (Fox et.al., 1995). PRIMAVERA developed advanced traffic management strategies for urban arterial roads. These strategies were developed with the aid of the NEMIS micro-simulation tool and then the best strategy was implemented on street. Much data was collected, firstly to calibrate and validate the initial micro-simulation model of the network, then to evaluate the effectiveness of the new strategies on-street. Data was collected for both AM and PM Peak periods. The full PRIMAVERA network in Leeds consisted of ten signalised intersections along 3km of an urban arterial, namely the Dewsbury Road, classified as the A653. This is one of the main radial routes into Leeds, carrying approximately 23,000 vehicles per day. It is also a heavily used public transport corridor, peak bus flows being in excess of 36 buses per hour.

To simplify the transferability tests carried out by the SMARTEST project, a sub-network of the PRIMAVERA network was used. This consisted of a 1½ km segment of the Dewsbury Road, containing two signalised junctions and a pelican crossing (see Figure 32). The test network also contains a number of priority junctions, where minor roads join the main arterial. Bus stops are also present in the network. The network thus contains many features that are common in urban road networks in the UK. An additional feature of the test network is that there is only one route between each of the origin destination pairs, therefore route choice is not an issue to cloud the model evaluation. It was also decided to only carry out simulation runs of the AM peak period.

Data Collected

Much data was collected during the PRIMAVERA project. In addition, a digital map of the area was available in AutoCAD format, which allowed the network geometry to be easily and accurately measured. The surveys carried out are summarised in Figure 33.

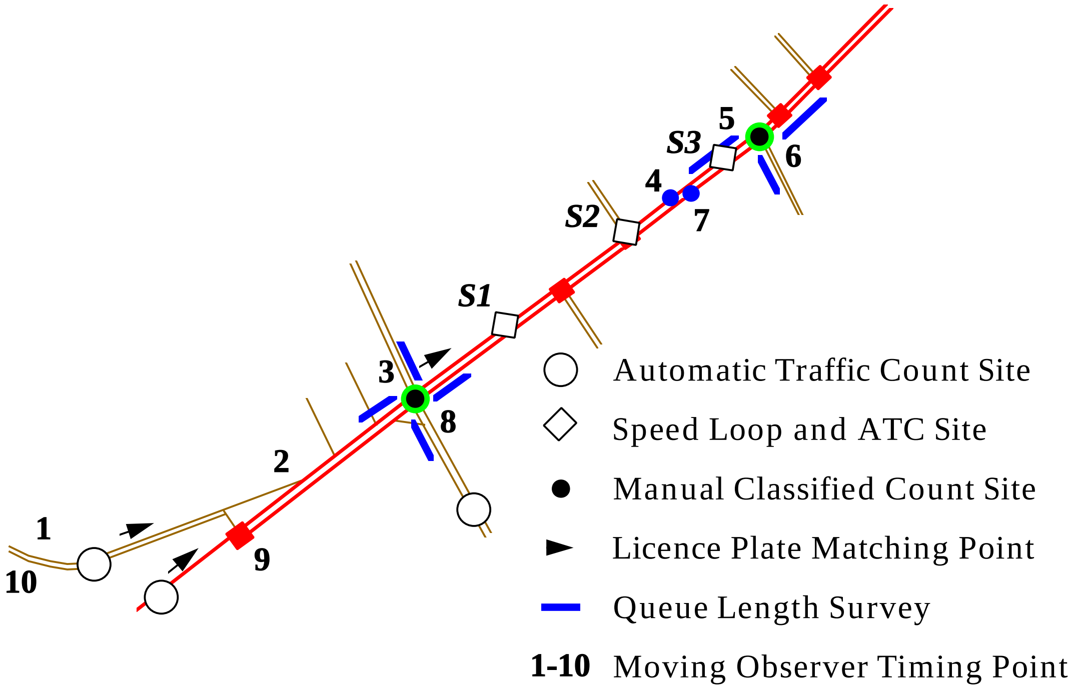


Figure 33: Field Trial Data Collection

Statistical analysis has been used to estimate the accuracy of the collected data. When comparing simulation outputs with data collected from the real world it is important to ensure that sufficient data has been collected to estimate the values being compared to a desired accuracy. If the usual statistical assumptions are made regarding the normality of the data then it is possible to determine the confidence interval for a population mean. The confidence interval is a range on either side of the sample mean. It is expressed as a function of a significance level, α, which usually has a value of 95%, and is given by the formula:

$$(t_L, t_U) = \left(\bar{x} - z_{1-a/2} \frac{s}{\sqrt{n}}, \bar{x} + z_{1-a/2} \frac{s}{\sqrt{n}} \right) \qquad (1)$$

where $z_{1-\alpha/2}$ is that value in the standard distribution that has 1-α/2 area to the left. For a 95% confidence level, $\alpha = 0.05$ and $z_{1-\alpha/2} = z_{0.975} = 1.96$.

The surveys on the PRIMAVERA network included:

- Automatic Traffic Counts, using loops, to measure traffic flows in both directions at three points in the network.
- Automatic Speed and Flow measurements as vehicles passed over at three data collection points on the Dewsbury Road.
- Manual Classified Counts at two of the junctions, to obtain turning movements for seven categories of vehicle.
- Travel time surveys both by number plate matching at three points in the network and by moving observers travelling in cars and buses around the network.
- Queue length surveys were carried out at the two signalised intersections.
- Bus waiting times at stops were measured by observers.

Modelling Approaches

All three micro-simulation models used similar network representations. Nodes represent junctions, and nodes are connected by links, each with a number of lanes.

Separate links are required for travel in each direction, i.e. none of the models allowed two-way movement on a link. This limitation can be important as it prevents overtaking via the oncoming lane if there is a suitable gap.

NEMIS has a minor limitation in that it can only model road networks where traffic usually drives on the right, so to model the UK network a mirror image has to be used. NEMIS also has a limit of four arms to a junction.

NEMIS is the only model that has provision for on-street parking.

AIMSUN2 has a very user friendly network builder that allows AutoCAD maps to be used as backgrounds. The road network model is then drawn over the top of this map. This allows an accurate network geometry to be specified without fear of error. AIMSUN2 was therefore the first model used to code up the Dewsbury Road network. The link lengths and their positions obtained from the AIMSUN2 model were then used to code up the NEMIS and DRACULA network models.

- *Car*-Following *and Lane Changing*

Driver behaviour is modelled via a car following rule and gap acceptance and overtaking rules. These usually have parameters which characterise desired headways, reaction times, aggressiveness, awareness and acceptable gaps for lane changing and turning across opposing traffic flows. Due to difficulties in measuring these parameters few of them are ever measured directly. The modeller relies on indirect measurements such as average headways, lane usage or saturation flow measurements to justify the values used.

AIMSUN2 uses a car following law based on that suggested by Gipps (1981) and a lane changing rule based on Gipps (1986). NEMIS uses a different car-following law, based on a study by Donati and Largoni (1976). Key parameters for four different vehicle types have been determined.

- *Vehicle Types*

Both NEMIS and DRACULA have a limit on the number of vehicle types allowed. DRACULA is limited to six types, namely Cars, Buses, Guided Buses, Taxis, High Occupancy Vehicles and Heavy Goods Vehicles. NEMIS allows seven types, namely five different types of private vehicle, plus buses and trams. AIMSUN2 allows multiple vehicle types to be specified.

Each vehicle type has associated with it a fixed set of parameters, such as acceleration and deceleration rates, vehicle length and car following parameters. Table 18 gives some of the default parameters provided for the various vehicle types used by each of the models.

DRACULA	Car	Bus
Maximum Acceleration (m/s/s)	2.5	2.5
Maximum Deceleration (m/s/s)	2.5	2.5
Length (m)	3.5	7.5

AIMSUN2	Car	Truck	Bus	Long Truck
Maximum Acceleration (m/s/s)	2.8	1.0	2.0	1.0
Maximum Deceleration (m/s/s)	4.0	3.5	3.0	3.5
Length (m)	4.0	8.0	9.0	12.0
Desired Speed (km/h)	90	70	60	70

NEMIS	All vehicles
Maximum Acceleration (m/s/s)	3.0
Maximum Deceleration (m/s/s)	5.0
Maximum Speed (km/h)	50.4

Table 18: Some default micro-simulation motion parameters

- *Public* Transport

The main drawback of AIMSUN2 is that it does not currently directly model public transport. Although it is possible to model a bus vehicle type, it is not possible to specify routes, timetables or bus stops. This can be very important in urban networks where it is often difficult for other traffic to overtake buses at stops. Buses can therefore have a significant effect on traffic flow in the network.

DRACULA and NEMIS allow both bus routes and bus stops to be specified. Both specify the routes by defining a list of links to be followed. Both use a start time and a generation frequency to produce the bus schedules.

For DRACULA bus stops are associated with bus services. For NEMIS the stops on a route can be used by any of services that use the route. Both allow multiple stops on a link. DRACULA uses a simple wait time model for the bus stops based on a passenger arrival rate, although this is not service dependent. NEMIS just has a stop time based on a sample from a normal distribution of a fixed mean and standard deviation.

- *Traffic Flows*

All three models have the ability to accept traffic flow data in the form of Origin / Destination (O/D) matrices. AIMSUN2 and NEMIS have built in route choice models. DRACULA uses the SATURN assignment model (Van Vliet, 1982) to calculate vehicle routes.

The vehicle generation models in DRACULA and AIMSUN2 assign an origin, destination and route to each vehicle as they are generated. NEMIS uses the results of its assignment model to produce turning percentages at each junction. So when a vehicle arrives at a junction, a random choice is made, based on the known turning proportions, to choose the direction the vehicle is to make.

AIMSUN2 is the only model that allows different O/D matrices for different vehicle types. This could be an important factor in the Leeds network, where HGVs have a slightly different O/D pattern to other vehicle types.

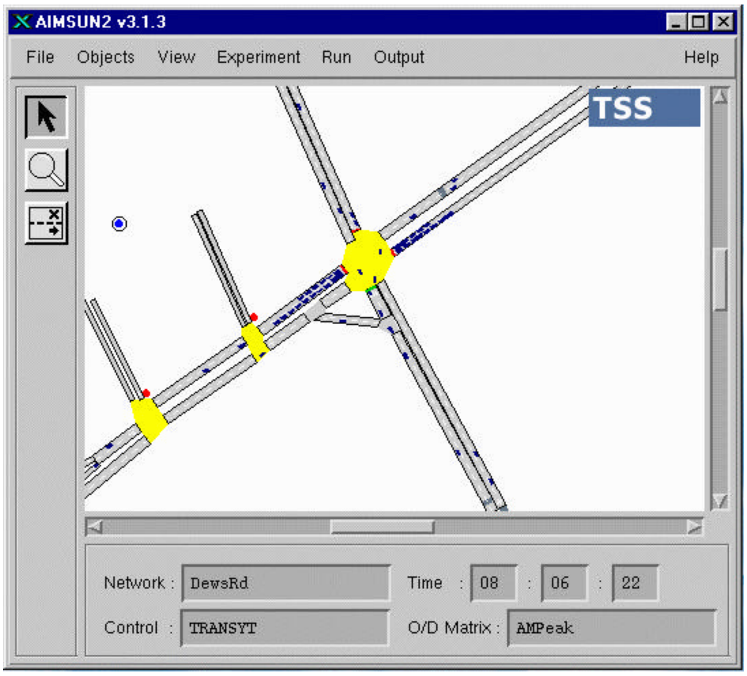
- Traffic *Signals*

All the models have the capability of modelling traffic signals operating under fixed time control.

The pelican crossing can be modelled as a two arm junction with 1 stage and a long intergreen. None of the models directly allow the demand response feature of the pelican crossing to be modelled (or other demand responsive features that may be present at other signalised junctions in the network). Pelican crossings only show red to the traffic if a pedestrian has pressed a button to register their desire to cross the road. AIMSUN2 does have the ability to allow an external module to be developed to control signals in the network so it would be possible to write such a module (as a dynamic link library) to model the correct actions of a pelican crossing. Time constraints have however not allowed such a development. It has therefore been decided that as during the AM peak it is likely that the pelican crossing will be used nearly every cycle, it can be modelled as if it was used every cycle.

Simulation Results

The Leeds network has been coded up for each of the three micro-simulation models. Calibration and validation has been carried out.



Network : DewsRd

Time : 08 : 06 : 22

Control : TRANSYT

O/D Matrix : AMPeak

Figure 34: AIMSUN2 simulating the Leeds network

Averages from four different runs using different random number seeds for each run were used.

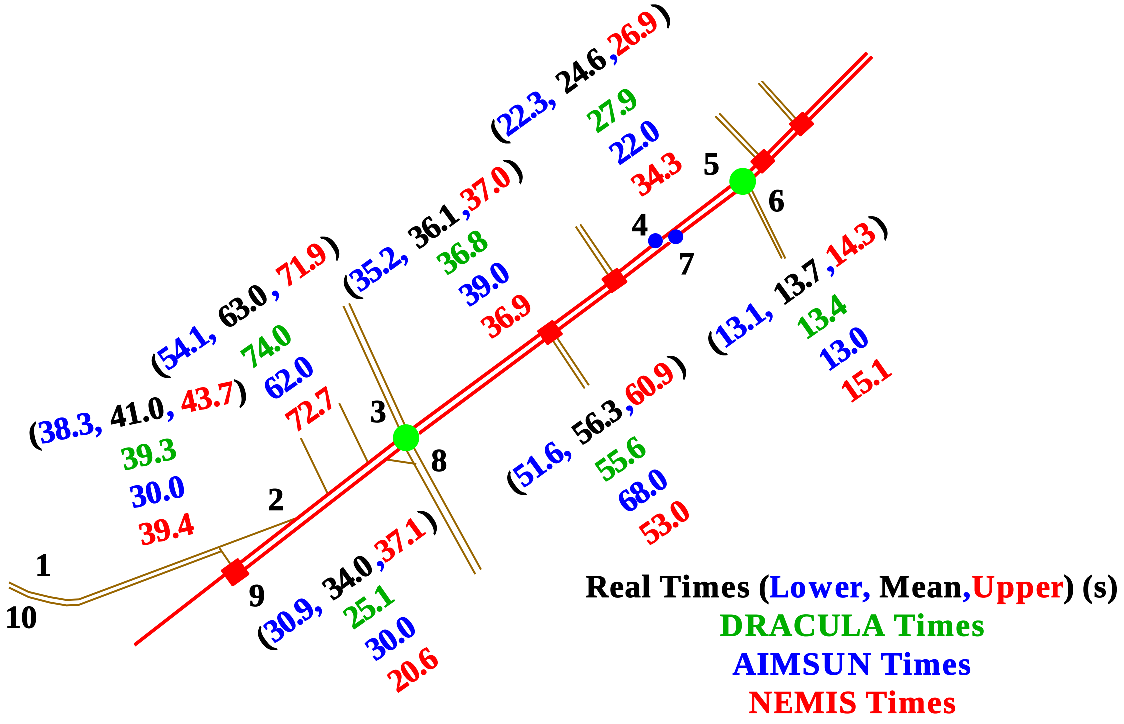


Figure 35: Link travel times from the different models

Figure 35 shows the link travel times from each of the micro-simulation models. These are compared with the actual travel times, as measured by moving observers and detailed in section 3.2.2. For each link, only the travel times for vehicles making the same turning movement at the end of the link as the moving observers were used in the analysis. As can be seen there is reasonably good agreement between the observed travel times and those output by the models.

Figure 36 shows a comparison of the queue lengths from each of the models. Here the agreement between reality and the model outputs is not so good. In particular the queue lengths from DRACULA and NEMIS are much longer than those observed for the Dewsbury Road link going into the Old Lane / Ring Road Beeston Park junction towards Leeds. This is a critical junction. The result indicates that both NEMIS and DRACULA have problems modelling junctions operating close to capacity.

Table 19 shows the times (in seconds) for each of the simulation models to model one hour in the AM Peak period. All the runs were performed on the same computer, which was a 200 MHz Pentium PC with 32Mb of memory. Runs have been carried out both with the graphics switched on and with them switched off. As can be seen, having animated outputs significantly slows down the simulation for all of the models. With the graphics switched off, both NEMIS and DRACULA are slightly faster than AIMSUN2. This can be partly explained by the fact that the AIMSUN2 runs were performed using the default step length of 0.75 seconds, whereas the DRACULA and NEMIS runs used a 1 second timestep.

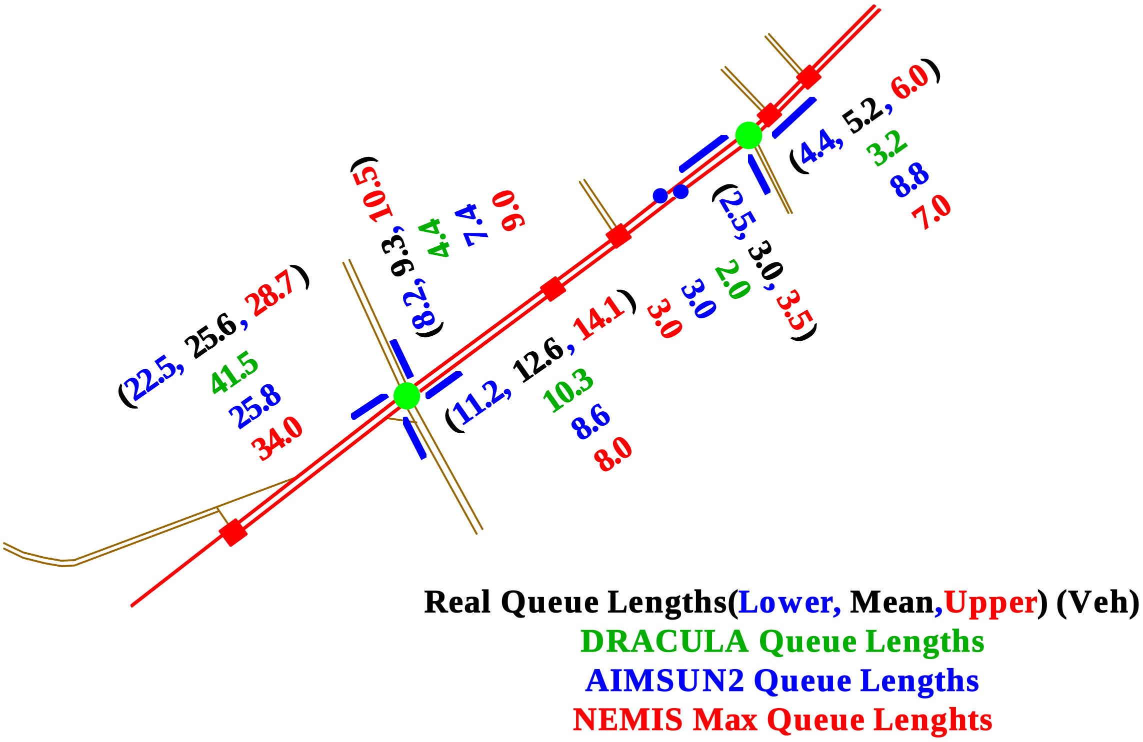


Figure 36: Queue lengths from each of the models

	Graphics on	Graphics off
DRACULA	68	24
AIMSUN2	375	39
NEMIS	64	24

Table 19: Simulation run-times (s) for one hour in the AM peak.

Conclusions

A variety of traffic data has been collected from a small urban road network in Leeds. This data has been processed and analysed so that it can be used in the calibration and validation of road traffic micro-simulation models.

Three micro-simulation models, initially developed to model traffic in different parts of Europe, have been used to model the traffic in the Leeds network. None of the models could represent all the features found in the test network, so some modelling assumptions had to be made to cover these cases.

The ability of the models to produce accurate representations of traffic behaviour has been investigated. All three models produce reasonably accurate outputs of travel times. Both NEMIS and DRACULA however have problems in modelling junction capacities accurately, which results in over estimations of queue lengths and travel times at junctions operating close to capacity.

AIMSUN2 in Stockholm (Swedish traffic)

The conclusions from this test were the following (based on AIMSUN2 version 3.2):

- A “weather parameter” is missing.
- In Sweden the driver’s choose the “right” lane (minimising future lane changes) as early as possible. In the model you can choose to change lane only on the section which is closest to the intersection.

- The on ramp behaviour could be improved. In most cities in Sweden you yield if you see a car approach from the ramp.
- A local parameter is missing. For instance in Sweden the capacity is higher on the ramp than on the motorway. This has been introduced in version 3.3.

General recommendations for calibration and validation

Network building

- If a digital map can be used then use it to help build the network.
- Try to avoid using short links where lane changing can take place.

Checking the basic model is correct

An animated display of the network in operation makes it easy to carry out basic checks that the model has been coded up correctly. Particular things that can be checked easily include:

- All the allowed turns at a junction are being used.
- Priority rules, give ways and stops, at junctions are being obeyed.
- Only public transport vehicles are using their reserved lanes.
- There are no unexpected queues anywhere in the network.
- Vehicle behaviour is appropriate for driving on the correct side of the road. This might not be obvious. It is easiest to always check that you have set the flag for driving on the correct side of the road appropriately.
- Signal phases and timings are correct. Observe each signalised junction second by second through a complete cycle and check that the phases are as expected. Then check the effective green times at a single signalised junction. If possible, choose a junction that is operating close to capacity, with little spare green time on any approaches. The effective green time at a junction is the proportion of the displayed green time that can sustain the saturation flow rate. In reality it is the displayed green time minus 2-3 seconds at the start of green while the flow rate rises from zero up to the maximum, plus 2-4 seconds at the end of the displayed green during the clearance interval, usually while the signal goes through amber to red. It is likely that the build up to the saturation flow at the start of green will be modelled correctly, it is not always the case that the cut off at the end of green will be. If necessary add two or three seconds of extra green time at the end of each phase so that movement through the junction at the end of the phase reflects reality.

Checking saturation flows

A key parameter when modelling road traffic networks is the saturation flow. For different streams of traffic passing through a junction, this is defined as the maximum flow rate that can be sustained by traffic from a queue on the approach used by the stream. It depends mainly on:

- the number and width of entry and exit lanes available to that stream and the effects of parked vehicles, bus stops etc. on lane width;
- the proportion of turning traffic and
- the radius of turn; and the gradient of the approach.

Traffic composition also affects saturation flows.

It is important to get the saturation flows correct when modelling a road network. The saturation flow effectively defines the maximum amount of traffic that can travel through a junction in any given time period. If the saturation flow is incorrect then estimates of junction capacity, throughput, delay and queue lengths will all be wrong. For many traffic models the saturation flow is an input parameter. The saturation flow is measured on street at each junction and the value obtained is input into the traffic model. For a micro-simulation model the saturation flow is an output. Its value depends on parameters used to define vehicle motion as well as the geometry of each junction. If the vehicle motion parameters have been well chosen then the saturation flows produced by the model should approximately agree with reality.

A method that has been used with some success is to measure on-street values of saturation flows at a junction in the network under study that is operating closest to capacity. Then adjust the reaction time in the micro-simulation model of the network until the observed flows through the junction agree with the on-street measurements. The flows through other junctions in the network then often agree with the observed values as well. Some micro-simulation models do not allow changes to the reaction time, or it is equivalent to the simulation time step, which can be difficult to change. In this case the other parameters, such as vehicle acceleration rates or minimum distances between stationary vehicles need to be changed in order to get good agreement between the model and reality. Unfortunately this can often result in clearly unrealistic values of these parameters being chosen.

Checking route choice

Use a traffic assignment package e.g. EMME/2 or SATURN to calibrate the OD matrix against observed traffic counts. The calibrated matrix may then be input directly into the micro-simulation model.

A more exact calibration could be obtained if a measured OD matrix is available in combination with measured flows and speeds.

Comparison between Macro and Micro Simulation

Introduction

In the Stockholm test-site, results from macro simulation as well as micro simulation runs are available. This provides an opportunity to compare the two different approaches, which may give an idea of the improvement in accuracy that can be expected from micro simulation modelling.

Incident management

Macro-simulation (EMME/2)

Frequency of incidents, remaining capacity, the delay in detection and reporting of an incident and the turn-out time of the rescue service are important factors when predicting the probable effects of improvements of Incident management strategies.

The method of calculating incident management with EMME/2 was simulating a set of traffic incident situations based on realistic frequencies for different incidents and the effects of reducing the duration of these incidents. The scope of incidents has been surveyed for the Stockholm region (Kronborg, 1993). As a result, the frequencies for different roads have been assessed in Table 6.1.

Table 20 Assumed frequencies for different types of incident (Lind, 1996).

Frequencies (per million vehicle-km)	Motorways	Country roads	Urban roads
- major accidents	0.6	1.5	1.5
- minor accidents	1.5	1.5	1.5
- vehicle breakdown (in a lane)	8	8	8
- other obstacles (incorrect parking, lost cargo, congestion)	5	4	8
- road works	0.8	0.4	0.4

The resulting delays were calculated using a queue model (QSIM) based on the details above (Jepsen, 1994). These delays were then used as additional link times in the EMME/2 assignment procedure. A major accident of 45 minutes primary duration, for example, leads to a total breakdown of the motorway. This incident will result in an average delay, according to the queue model, of 22 minutes if it occurs in the middle of the day (11:00 am) and 53 minutes during the peak morning hour (7:30 am). Consideration was given to traffic flow, speed limit, bottleneck capacity, starting time and duration.

To simplify the calculation process, the rather complicated system above was simplified and QSIM calculations only made in the cases in Table 1.2.

Table 21 Capacity and duration for QSIM calculations (Lind, 1996a).

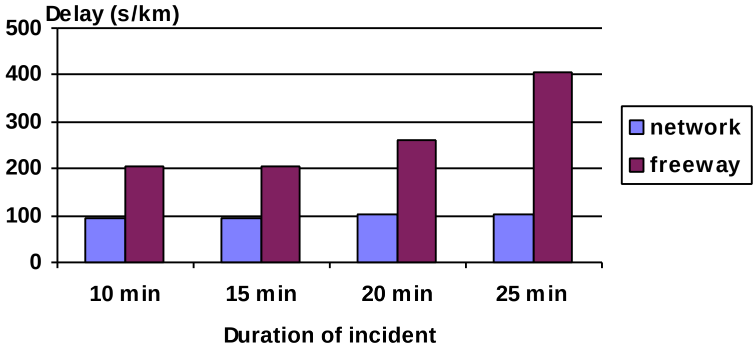
	capacity	duration
- major accident	0%	45 minutes
- minor accident	40%	30 minutes
- vehicle breakdown (in lane)	75%	20 minutes
- other obstacles	67%	30 minutes
- road works	80%	24 hours

To calculate the probable effects of incident management, the delays associated with individual incidents of varying duration have to be simulated.

Micro-simulation (AIMSUN2)

The main parameter to model incident management using micro simulation seems to be the duration of the incident. The duration of the incident reflects the time it takes to detect the incident, report it, process the information and clear the incident. A sensitivity analysis was made where the delay on the motorway and the network as a function of incident duration was simulated. Figure 1.1 clearly shows the non-linear growth in delay and the corresponding importance of reducing incident duration.

Figure 37 Incident management in AIMSUN2.



The effect on delay due to the incident seems to be modelled in a reliable manner, as far as can be judged from the animation of the simulation run. If the 20-minute incident specified is increased by 25 percent (5 minutes), the average delay for the vehicles using the freeway is increased by 60%. This is in reasonable accordance with queuing theory that says that the delay increases quadratically with incident duration. There is evidently a great potential in quicker incident clearance under these traffic conditions.

The reduction in capacity due to an incident is modelled as a blockage of a discreet number of lanes. It is not possible to specify a certain percentage of capacity reduction as in the macro-simulation, which would be useful if for instance half a lane was blocked. There are also two phenomena concerning incidents that are not modelled in AIMSUN2. The first is the fact that vehicles in the lanes that are not blocked by the incident or the queue drive slower than normal. However, this behaviour is going to be implemented in the next release of version 3.2. Secondly, the phenomenon of ”rubber necking” is not taken into account i.e. cars going in the opposite direction slow down to look what has happened.

Apart from these shortcomings, there seems to be good possibilities to model incident management in AIMSUN2.

On-trip information and VMS

Macro-simulation (EMME/2)

The method of calculation used for on-trip information was based on the simulation of a traffic system subjected to some form of interference. It took into consideration realistic frequency levels for different incidents in the traffic system and the effects resulting from the road-user being better informed, thus allowing new route choices to be made based on this information. As stated below, a range of factors must be assessed in order to calculate the effects of traffic information. Among other things, the quality of information must be specified and the behaviour assessed in the light of the expected reliability and timeliness of the information.

The efficiency of the traffic information system may be considerably reduced if the information chain from detection to reporting is not sufficiently rapid. If this is the case, many road-users may have driven straight into the traffic jam before the information reaches them. To assess this factor properly in the case of on-trip information, it is important to estimate the joint probability of detection and reporting.

Table 22 Reporting probability with on-trip information (Lind, 1996a).

Probability	Motorways	Rural roads	Urban roads
- major accident	90%	70%	80%
- minor accident	50%	25%	30%
- vehicle breakdown (in lane)	10%	5%	20%
- other obstacles	15%	5%	10%
- road works	90%	60%	70%

Automatic detection systems are used mostly on motorways and mobile telephones mostly in rural areas. The detection time was estimated to 3-5 minutes on a motorway, 5-10 minutes in an urban area and 15-30 minutes in a rural area. The processing time at the traffic control centre prior to information being issued was assumed to be 2-5 minutes. Information is thus sent out via VMS and RDS 5-35 minutes after the incident has occurred.

Traffic not affected by the incident was assigned according to the static equilibrium as interpreted by the EMME/2 system. The on-trip information was assumed to be provided as disruption information (location, probable duration) and alternative routes. It was assumed that only main roads were used in the regional traffic information as misunderstandings could easily occur otherwise.

Incidents were assigned at random to the road network based on disruption frequency data. The probability was assumed to be proportional to the number of veh.-km on each link.

Micro-simulation (AIMSUN2)

Modelling on-trip information

AIMSUN2 contains a route choice model for dynamic re-routing. At the start of the simulation, after the warm-up period, new shortest routes are calculated based on simulated travel times. If the dynamic option is chosen, new shortest routes are calculated during the simulation with an interval specified by the user. This means that for each time interval there is one specific path between two points in the network that is the shortest according to the simulated travel times.

For guided vehicles, information on new shortest routes is given every time interval and hence they may change their route during the trip if the dynamic option is chosen. The program saves information about which routes have been the shortest during the last intervals, and the model distributes the guided vehicles among these routes according to a logit model. This means that not all guided vehicles use the present shortest routes but some stick to the older shortest routes. This gives certain inertia to the routing behaviour.

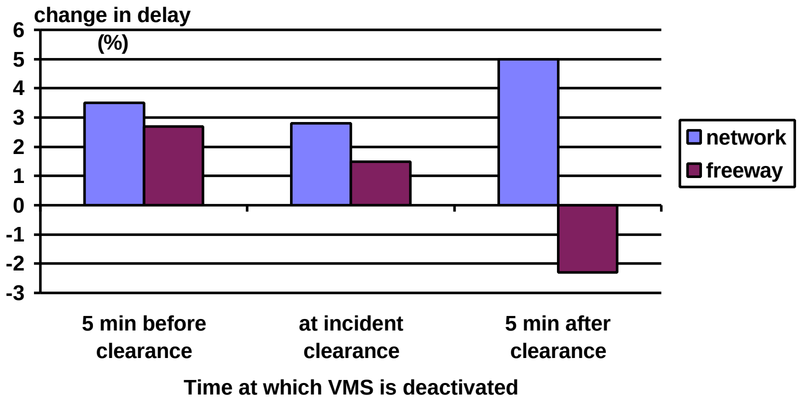
In summary, the following parameters can be used to tune the dynamic route choice:

- Fixed or variable routes
- Capacity weight factor for initial routes
- Proportion of guided vehicles for each vehicle class.
- Interval at which new routes are calculated.
- Type of model to distribute drivers between routes, logit or binomial
- Number of routes to consider in the logit model
- Scale factor for weighing the routes in the logit model
- Modelling VMS

Variable message signs can be modelled directly in AIMSUN2. The way to work with VMS differs depending on the type of demand data used as input, OD matrices or input flows and turning percentages. Since OD matrices are used in this evaluation, the description below refers to these more elaborate re-routing capabilities for modelling VMS in AIMSUN2 provided when using OD matrices.

In the SMARTEST Stockholm test network, there is a route choice between the freeway (western route) and the eastern arterial. In order to inform the drivers of the incident conditions, a VMS is positioned at intersection 1 (Järva Krog) visible for drivers heading south towards the downtown area or continuing on the southbound freeway. The chosen message is ”Accident at Haga Norra” referring to the incident scenario where a truck blocks the freeway at intersection 3. The behavioural assumption connected to this message is that 50% of the drivers with destinations in the southern urban part of the network divert at the location of the VMS and choose the alternative route. The sign is lit 3 minutes after the incident occurs, i.e. there is a delay that reflects the time it takes for the incident to be reported and processed by the traffic management centre. Then the message is active until the incident is cleared. In order to study the significance of the time the message is active, two additional simulations where run: the sign is turned off five minutes before the incident is cleared and five minutes after. Figure 6.2 shows the resulting delay per kilometre in the different cases.

Figure 38 **Variable Message Signs in AIMSUN2.**



The graph shows that there is an *increase* in delay on the network level due to this VMS strategy. As a whole, the impacts are quite complex and must be divided into sub-effects to be understood:

- When turning the sign on, a substantial portion of the drivers start changing lanes in order to make the turn to the off-ramp. This behaviour creates a disturbance in the freeway flow and the delay increases. This effect is probably exaggerated in the model and explains why there is an increase in delay on the freeway when the sign is activated for only a short time.

- If the sign is active for a longer time, the delay on the freeway decreases due to the fact that fewer drivers are stuck in the queues.
- Since the network delay increases, it is obvious that the alternative route is not better than the original despite the incident.

In conclusion, this example of VMS application was not very successful. If the incident impacts on the freeway traffic had been greater, there would possibly have been a positive effect of re-routing by VMS. Instead, it can be seen as a successful application of a VMS simulation – sometimes the result will be that VMS signs are not a good strategy.

Provided that the above factors are considered, there are good possibilities to model VMS effects in AIMSUN2.

Conclusion

In order to study local capacities of links, intersections and ramps or control strategies for Incident management and On-trip information, micro-simulation is very useful. The major problem up to date concerning ITS is however to get reasonable behavioural information to represent various ITS applications. For the innovative user, macro-simulation still offers good possibilities to model average network effects, as the behavioural assumptions seem to be more decisive than the modelling detail. In this case, results from micro-simulation can be used to produce input data to assignment models. In the long run, however, micro simulation offers better possibilities than macro-simulation to model various dynamic phenomena.

CONCLUSIONS

The SMARTEST project has successfully completed and has achieved its objectives.

1. A review of existing micro-simulation models has been carried out and a State-of-the-Art review report has been produced (SMARTEST Deliverable 3).
2. The project has investigated how the existing models can best be enhanced to fill the identified gaps, thus advancing the State-of-the-Art. A requirements specification (SMARTEST Deliverable 4) has been produced to detail what the identified gaps are. A design specification (SMARTEST Deliverable 6) has been produced to detail how these gaps were filled by enhancing the four micro-simulation models under development by the project partners.
3. The four micro-simulation models (AIMSUN2, DRACULA, NEMIS and SITRA-B+) under development by the project partners have been enhanced according to the needs of users in Europe.
4. A best practice manual for the use of micro-simulation in modelling road transport has been produced (SMARTEST Deliverable 8), which also includes results of a transferability study to check that the models can be used at different locations in Europe.
5. The findings of the project have been widely disseminated throughout Europe, via appropriate publications, conference presentations and through the project's World Wide Web pages at http://www.its.leeds.ac.uk/smartest.

By sharing experiences and data sets, an enhanced set of micro-simulation tools have been developed which improve on the State-of-the-Art, and which are transferable across Europe. Confidence that the tools have been correctly validated has also improved. The tools developed in this project can produce outputs for a wide range of performance indicators, allowing any European scheme objectives to be evaluated.

The SMARTEST project has provided road network managers with an improved set of tools and procedures to assess the impact of road transport schemes and interventions. Road network managers supplied with such a set of tools can make considerable economic savings as they will be able to accurately assess new schemes without the expense of field experiments. Such assessments can also demonstrate the usefulness of improved UTC and information and guidance systems and hence lead to new industrial developments. Improved evaluation of technical innovations and operational strategies on the road network will result in improved efficiency of operation of the road system improving the chance of optimisation of the transport networks. Better assessment of safety and environmental impacts will allow policies to be developed that reduce accidents and pollution. Better micro-simulation packages will also improve traffic, transport and information management. This will result in better knowledge and understanding of mobility, traffic flows, their interactions and interdependencies.

LIST OF PUBLICATIONS

CONFERENCE PRESENTATIONS

Presentations of the SMARTEST project have already been presented at the following conferences:

ISATA. Florence, Italy. June 1997.

Third World Congress on Intelligent Transport Systems. Berlin, Germany. October 1997

EURO Working Group in Transportation. Autumn 1997

INFORMS Meeting, Montreal, Canada, April 1998

European Transport Forum (PTRC). Loughborough, UK. September 1998.

TRB. Washington, USA. January 1999

Universities Transport Studies Group, York, UK, January 1999.

Presentations of the SMARTEST results are also planned at the following conferences:

European Transport Forum (PTRC). Oxford, UK. September 1999.

Fifth World Congress on Intelligent Transport Systems. Toronto, Canada. November 1999

PUBLICATIONS

The following paper has been accepted for publication:

- Dougherty, M., Fox, K., Cullip, M. and Boero, M. Technological Advances Which Impact on Micro-Simulation Modelling, Transport Reviews (1999)

Papers which are planned for submission to journals include:

- Review of micro-simulation models – Transport Reviews
- Transferability of micro-simulation models – Transportation Research
- Calibration and validation procedures for micro-simulation models – Transportation Research
- Will ITS Work For You? The SMARTEST project has been developing tools to help you find out – Transport Technology International
- New tools to evaluate Intelligent Transportation Systems - the SMARTEST project - Traffic Engineering and Control

WORLD WIDE WEB

The SMARTEST home page can be found at:

http://www.its.leeds.ac.uk/smartest/

The main sections of the Web pages are described below.

Project Summary. A summary of the project is given, including the project objectives, main expected deliverables, and links with other projects, tasks, areas, programmes and policy actions.

The Partners. Contact details are given for all the partners in the SMARTEST Consortium. This includes: Contact Name, Address, Phone Number, Fax Number, E-Mail address, WWW pages.

The SMARTEST Tools. A description of the micro-simulation tools (AIMSUN2, NEMIS, DRACULA and SITRA-B+) being enhanced within the project is presented.

The Deliverables. A list of all the project deliverables is displayed. As soon as each public project deliverable has been accepted by the Commission it is placed on the Web Site. Acrobat, Word and HTML Versions of each deliverable are produced to ensure maximum dissemination. The Review of Micro-simulation Tools and the Best Practice Manual are both project deliverables.

Micro-simulation abstracts. The review of micro-simulation tools carried out in Workpackage 2 revealed many papers on micro-simulation. The abstracts of these papers are presented on the SMARTEST web page to allow researchers to track down detailed information about the state-of-the-art of micro-simulation.

Micro-simulation links. A search of the World Wide Web has uncovered a number of web pages relating to micro-simulation. Links to these web pages are provided on the SMARTEST web site.

MICRO-SIMULATION WORKSHOP

At the end of the project a SMARTEST micro-simulation workshop was be held in Toulouse. Its purpose was to review findings and discuss the results of the project with users and researchers who use simulation tools in assessing the benefits of ITS applications and traffic management schemes.

The workshop had the following content:

- User requirements concerning traffic management
- Model development within the SMARTEST project
- Evaluation results, with examples from test-sites
- Discussion of recommendations and conclusions

Users from different cities in Europe as well as consultants and researchers not involved in the SMARTEST project were invited to discuss the recommendations.

The proceedings of the workshop are available as SMARTEST Deliverable 9: Micro-simulation Workshop.

REFERENCES

DONATI, F. AND LARGONI (1976) Analisi del comportamento di una colonna di autoveicoli in condizioni perturbate, *Riunione Annuale AEL*, Sorrento, 1976.

FERRER, J.L. AND BARCELO, J. (1992) *Modelos microscopicos de simulacion de sistemas de guiado de veh'culos*. Research Report. Laboratori d’Investigació Operativa i Simulació, Universitat Politècnica de Catalunya.

FOX, K., MONTGOMERY, F.O. AND MAY, A.D. (1995a) Integrated strategies for urban arterials: DRIVE II project PRIMAVERA. 1: Overview *Traffic Engineering and Control*, **36**(5), pp268-271.

GIPPS, P.G. (1981) A behavioural car-following model for computer simulation. *Transportation Research Board*, **15-B**, pp. 105-111.

GIPPS, P.G. (1986) A model for the structure of lane-changing decisions. *Transportation Research Board*, **20-B**, pp. 403-414.

LIU, R., VAN VLIET, D. AND WATLING, D.P. (1995) DRACULA - Microscopic Day-to-Day Dynamic Modelling of Traffic Assignment and Simulation, Paper presented at the *Fourth*

International Conference on Applications of Advanced Technologies in Transportation Engineering, Capri, Italy, 27-30 June 1995.

MAURO, V. (1991) Evaluation of dynamic network control: simulation results using NEMIS urban micro-simulator *Transportation Research Board Annual Meeting*, Washington DC.

VAN VLIET, D. (1982) "SATURN - A Modern Assignment Model", *Traffic Engineering and Control*, **23**, pp578-581.