ALMA MATER STUDIORUM - UNIVERSITY OF BOLOGNA

FACULTY OF ENGINEERING

MASTER DEGREE OF CIVIL ENGINEERING

DISTART - TRASPORTI

FINAL DISSERTATION

in
INFRASTRUCTURE SYSTEMS

MICROSIMULATION OF TRANSPORTATION SYSTEMS
THEORY AND APPLICATIONS

EXAMINEE
Claudio Borsari

SUPERVISOR:
Chiar.mo Prof. Joerg Schweizer

CO-SUPERVISORS:
Ing. Diego Deponte
Ing. Caterina Randone

Academic Year 2011/2012

Session I

To my granma,
for growing me up

To my family,
for forcing me to learn English and allowing me to travel the world

To arch. Fabio Casiroli, Diego, Caterina, Elisabetta
and all the Systematica team
for everything I’m learning from them.

TABLE OF CONTENTS

1. INTRODUCTION

1.1. From Macrosimulation to Microsimulation Models

Cities and towns experience a continuous traffic growth. At the same time, municipal governments lack the funds to meet the new demand with new infrastructure: this traditional approach showed all its limits because of the restrictions of available budget and the increase of infrastructure cost, especially in high-density central areas (where traffic concentrates). The presence of new infrastructure, moreover, attracts new traffic, thus creating a vicious circle where new infrastructure is needed. This is the so-called “induced traffic”, i.e. new traffic created by the new and better connection between areas not connected before. The new traffic, then, saturates the new infrastructure, calling for a new one. Then the process starts again.

Because of the failure of this solution, that we experience every day, new approaches to the problem were tried: the most important are the traffic (or demand) management and the integration between planning and operational design of roads.

The first approach is more on the planning side: it is about planning cities in such a way that the need for car trips is reduced to a minimum, placing the most important traffic attractors along the most important public transport corridors or at public transport interchanges: this will make the destination easily accessible from many public transport routes with no changes, making the public transport travel more attractive. Moreover, this approach is based on mixeduse zones, opposed to the single destination (residential or commercial or office) zones, in the attempt to distribute services and workplace in the city territory to reduce the travel distance from households, making more attractive mode choices such as cycling or walking.

This thesis will focus on the second approach: the integration of planning and operational design to improve the road performance and the total efficiency of the system. This approach has been made possible by the widespread availability of powerful computers and by a new generation of software, whose origin dates back to 15-20 years ago. The main goal of this approach is to maximize the performance of existing road systems without the addiction of new infrastructure, but managing and improving the information to drivers, traffic signals, intersection management and the use of road space.

The new approach requires new models: from the traditional macro-simulation models, the focus is now on micro-simulation models, with the attempt of merging the advantages of the two in the meso-simulation models.

Macro-simulation provides an aggregated representation of the demand, expressed in terms of total flow. The model behind this kind of simulation is a classic 4-step model, based on trip generation (the generation and attraction of each zoned is transformed in a comprehensive list of trips from an to each zone), trip distribution (the trip generated from each zone are distributed among zone pairs, called Origin/Destination pairs), mode choice (for each O/D pair, trips are distributed among the different modes of transport available on that route according to the utility assigned to them by travelers, given the purpose for traveling) and assignment (traffic loads are distributed on routes, minimizing the total costs). The result of this process is a set of hourly flows on links, following the stochastic user equilibrium, which optimizes the driver's utility, within the limits of the stochastic preferences of the operator.

Meso-simulations split the time and space structure of macro models, hours and links, into discrete time intervals and sections respectively. The main parameters to model traffic are now platoons, opposed to flows, density, flow and speed, the latter being correlated by the fundamental equation

$$Flow = Speed * Density$$

Meso-simulations work according to conservation models of traffic density, coupled with priorities, constraints and traffic rules of behavior from a finite set of activities (join, cruise, change lane). Vehicles are split among different O/D pairs and each group is divided in different vehicle types. Each type has an activity plan, i.e. a set of probabilities of engaging in a maneuver (join, cruise, change lane, etc.) in the following time interval. The activity plan may vary in space and time. The traffic is updated following these steps: first, time and space network constraints (each maneuver requires space and time, space may vary with speed) are applied to maneuvers commanded by the activities; then an estimation of completed maneuvers is calculated and the maneuvers for which there is not enough space in the section are rejected according to the priority rules. An intermediate state is calculated, assuming that the maneuvers (lane changing, join, split, change from platoon leader position to follower position, etc.) are completed without forward movement of traffic. Finally, allowable speed according to the space in the downstream section is calculated and the traffic flow moves according to that flow. The output of these model is a flow rate per hour.

Micro-simulation models focus on single vehicles. Vehicles and drivers are defined by a set of properties that determine their behavior, such as aggressiveness, awareness, gap acceptance, minimum headway from the previous vehicle, etc. These properties are assigned to the population following statistical distributions and perturbation (a variation in the statistical distribution that allows for a different perception of costs among user within a limit set by the operator). Micro-simulators can as well model in detail the road geometry and characteristics, intersections, traffic signals, number of lanes, etc. There are many possible outputs: information on queues, flows, maneuvers, emissions, etc. Micro models cannot simulate modal changes: public transport and pedestrians are simulated just in their interaction, in terms of delay, with other vehicles. If I change the public transport service in a micro-simulation network, this will impact the traffic flow but not the demand, i.e. it is impossible to foresee, using this software, the effects of modal share of public transport changes. This is anyway out of the scope of this kind of simulation: they exist only to manage and improve the road system performance in the short term with punctual interventions and not with large strategic projects.

1.2. Demand Models

The Demand Models are not, strictly speaking, a part of any simulation model, macro, meso or micro. They represent a mean through which a disaggregated amount of socio-economic and geographic data come together to define the number of trips generated and attracted from each zone of the study area. The process can be divided into three parts: trip generation, trip distribution and modal choice.

The Trip Generation starts from the analysis of population and activity distribution throughout the study area and, in general, census information. The first set of data is used to define the zones, smaller where there a more precise evaluation is needed (high concentration of population and activities in urban areas) and larger in less populated zones. The precision of the model is related to the dimensions of the zones because transport models do not account for trips internal to the zones: because of the small distance generally covered in urban trips, with large zones a relevant number of them would be lost. On the opposite side, the number of trips in countryside areas is smaller and trips are generally longer, so that it is possible to increase the dimensions of the zones.

Population surveys (such as census surveys) are used to define the purposes of the trips and the number of trips generated, according to the socio-economic condition of the considered population. The population can be divided by age, family status and car ownership. According to this and other information, the number of trips generated by the zone are generated.

Each zone will also have a part of attracted trips, the number of people that reach that zone for any purpose. This attraction capacity is related to the purpose of the travel through many different parameters: as an example, commuting trips will be directed to zones with availability of jobs (given the average surface per worker parameter), study trips to schools and university (given the average surface per teacher parameter and the average number of students per teacher), business to tertiary and office districts, and so on. These parameters are then multiplied to the total surface per function in the zone to obtain an estimation of the number of trips attracted by the zone. The more parameters and trip purposes are included in the model, the more the estimation will be precise.

Once the generation and attraction, for each zone and each purpose, are known, it is possible to start the following step in the demand generation, the Trip Distribution: this is the calculation of the most probable spatial distribution of trips, by purpose, between zones of origin and zone of possible destination, considering, for each O/D pair, the generation level of the zone and its relation with neighboring zones generation level, the attraction level of the zone and its relation to the attraction level of neighboring zones and the generalized cost of the path between origin and destination.

Generalized cost is a quantification of the convenience the user gives to each possible route to reach its destination. Different generalized costs are applied to different transport modes. For private transport, in general the cost has three factors: value of time, distance (including fuel and vehicle operating costs) and other costs, such as parking fees or tolls. Public transport generalized costs are more complex, and include factors to quantify the value of in vehicle time, waiting time, transfer penalty and walking time, beside of course to the public transport fare. Specific surveys and estimations on costs are needed to complete this part.

Another factor influencing the relation between O/D pairs is the impedance factor. This is a factor, proportional to the distance between the origin and destination, that reduces the attractiveness of the trip for the user. This reduction highly depends on the purpose of the travel. As an example, the AMAT (Agency for Mobility, Environment and Land Use) model for the city of Milan shows that most of the non-work and non-study trips are shorter than 2.5 km, while work and study trips often reach length of more than 5 (work) or 15 (study) km.

Once the trips have been distributed, through interviews and surveys it is possible, for each trip purpose and type (in Milan total trips are divided in Milan to Milan, external to Milan, Milan to external) divide the total number of trips among the different hours (or time periods, such as AM/PM peak period, AM/PM peak hour, inter peak) to obtain a daily distribution of trips, useful for the next phase of the demand modeling, the modal choice.

The modal choice couple each trip to a transport mode according to the most convenient mode for that particular trip. The most important factors in the modal choice model are: purpose of the trip (frequent or systematic trip, trip due to the need to transport something, etc.); time of the trip (not all modes are available at all times); travel time, cost and accessibility of each mode with respect to the trip; trip chain of non-home based trips (route), conditioning the mode choice of successive trips in the chain because, obviously, if I leave home by bus, I will not have a car available for the following trips. There are many, the most common is the Logit model: the probability of choosing a transport mode depends on an utility factor influenced essentially by the value of time for each kind of trip. An evolution of this model is the Nested Logit, that is divided in two steps, the first differentiating the trips between public and private transport, the second dividing the public transport trips among the different available modes.

The next and last phase is the Assignment phase, but this is the core of transport simulation modes (for all the kinds of model, macro, meso and micro) and will be discussed in Chapter 2.2, specific for the Assignment problem.

Each step described before should be calibrated: “Calibration is the process of adjusting the parameters used in the various mathematical relationships within the model to reflect the data as well as is necessary to satisfy the model objectives” (DMRB Volume 12). The Calibration process compares obtained model outputs against observed data used to build the model, looking for errors in the equations’ parameters. It is important to calibrate each step, especially in large and complex models, to find errors easily: it is easier to check one step at a time and proceed with the model building being sure that the previous steps are correct, than find errors only at the end of modeling process with a lot of calculations and steps to review. Calibration can be done on network (changing global or local parameters, such as speeds, response times of drivers, visibility on links, etc.) or during the OD matrix development or on the trip assignment working on cost functions.

1.3. Research Developments

There are two main research fields on demand models and the use of simulation levels that are currently being investigated: the Variable Demand Models and the integration of micro-simulations into meso-simulations to analyze the wide area impacts of punctual modifications on the road systems, especially in terms of re-routing.

The main difference between a traditional model and a Variable Demand Model (VDM) is that the route costs after assignment do not influence only the modal choice phase, but are connected, in a loop, to the trip distribution or even the trip generation phase. The phenomenon that this new kind of model tries to recreate is that a zone is far less attractive to people if its connections are jammed with traffic, increasing the travel time and costs. People will look for alternative destinations with the same functions. On the generation side, residents in an high traffic area will try to minimize their travels if they cannot avoid the congestion, or they will look for a different route or transport mode. An example of VDM related with a micro-simulation model is the S-Paramics model of Chippendale, Wiltshire, UK. The project integrated the S-Paramics software in a public transport and parking demand assessment following new developments in town. S-Paramics has been used to calculate with precision the buses travel times related to their interaction with other vehicles in the congested network. The output of the micro-simulation was then used as input to update the demand for public transport, not only in terms of quantity, but also in terms of trip re-timing, by the population, to maintain the same arrival time. This allowed to re-allocate trips not only at a macro-scale (from a time period to another) but also at a micro-scale, for example a 10 minutes anticipation of the trip due to the knowledge, by the user, of congestion problem and probable delays on the bus service. This kind of simulation allowed S-Paramics to deal with a multimodal model and to be a helpful tool in the analysis of traffic impact of public transport variations.

Another interesting application of an hybrid micro and meso simulation comes from Stockholm. Since a micro-simulation gives very good results on very small areas and allows to analyze in detail the effect of different traffic signal timings, and a meso model is quicker to build and use to obtain traffic analysis on a wide area, the attempt is to simulate one or more small areas of the network with micro-simulation, then analyze the large scale effect of changes with a meso-simulation model, such as redistribution of routes over network due to improvement to a single intersection and effects of the new situation on O/D pairs. Also, the continuous communication between the micro and meso models allows for better input data in the micro models. The connection between the micro-model and the meso model are coded through virtual links that distribute traffic from the aggregated network representation on MEZZO (the software used in Stockholm for meso-simulation) to the detailed representation on VISSIM (micro-simulator). This allows for queues to propagate upstream from the micro-simulator to the meso network following the density calculated in the micro-simulations, and, on the other side, the cars approaching the micro model boundaries gradually adapt their speed to the average speed of the large network in the meso-simulation and are aggregated into platoons. Future application of this hybrid modeling, as stated by the authors of this research, include: a better calibration of meso models to achieve more reliable flows and a more realistic simulation at a meso level of bus operation, adding bus services, timetables and bus lanes.

2. METHODOLOGY OF MICROSIMULATION

2.1. *Input Data*

The data obtained from the demand estimation models illustrated in the previous Chapter are used as inputs in the micro-simulation models in different ways. The first approach is more observation-oriented and requires the user to define, within the micro-simulation model, the flow on each existing itinerary in the study area. The second approach uses as input the available O/D matrices.

2.1.1. *The Itineraries Approach*

This first approach requires the user to input in the model the volume and traffic composition (light vehicles, heavy vehicles) on each link. Once the flow has been defined, all the possible itineraries must be defined: at each intersection and for each link, an origin (section of choice on the link, generally close to some kind of intersection) will be connected to a set of possible destinations. Each itinerary will then have defined a relative volume, or a percentage of the total flow, that will make that particular choice. An itinerary is defined as a sequence of links connected by these choices. The definition of demand through this method does not need a system of zones, because the itineraries start and end on links.

This is a static assignment: there is no cost function, the flows are exactly the ones observed in reality and no route choice is possible. It is possible to simulate the effect of changes in network situation (a new timing for a traffic signal, for instance) on the existing flows, but since there is no route choice, it is impossible to see the effects of re-routing due to congestion and the subsequent increase of travel cost of the affected areas.

This approach is very time consuming for large networks and requires a large amount of observed data, or data coming from a large scale model, such as a meso or macro simulator (see the Stockholm case in Chapter 1.3).

2.1.2. The O/D Matrix Approach

This approach needs the definition of a zone system to which an O/D matrix is assigned. The source of the matrix can be a macro or meso model or a traffic survey. In the first case the matrix is already calibrated and can be a direct input for the model. It may be necessary to analyze the correspondence between the links defined in the two different models on the boundary between the respective study areas: because of the scale, in the macro or meso models not all the links are represented (sometimes not even in the micro simulation is represented every single real link). A single link in the macro model can be represented by more than one link in the micro model. In general, the software automatically distributes the generation from a zone among the different links leaving the zone according to their characteristics and the destination of the vehicles, if more than one link is present. Thus, the zone representing the demand from the macro model can be defined as covering more than one link in the micro-simulation. Another option has been explored in Stockholm (see Chapter 1.3), connecting one or more micro-simulation links with a single meso-simulation link with virtual links as shown in the figure below.

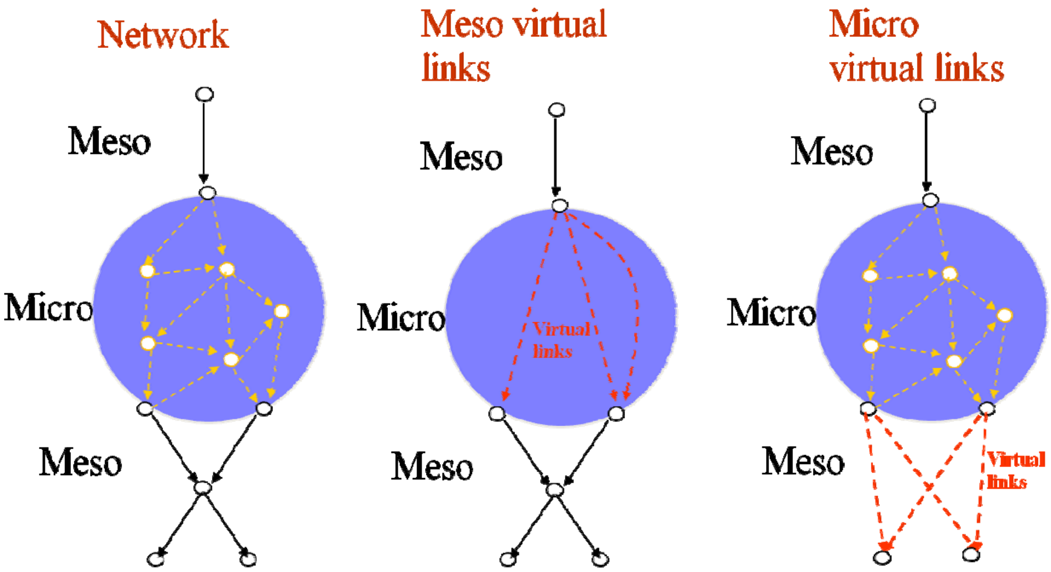


Figure 1: scheme of virtual links connecting a meso and a micro simulation

If an O/D matrix from a macro model is not available, traffic surveys are needed to collect the needed data. This observations can be used to update an old O/D matrix from previous traffic studies, or, in the worst case, these data can be used to create a new matrix from scratch.

In general, the matrices derived from traffic studies are not complete and may contain gross errors in case of networks with route choice possibilities. If the network is simple, i.e. it is a single intersection verification problem, the observations are enough to build a complete O/D matrix. In this case it is easy to calculate the O/D pairs starting from the observed flows at each arm of the intersection and the flows of each maneuvers (when surveying an intersection, all the maneuvers are counted). If the network is larger than that, it gets more difficult to survey each possible itinerary and maneuver, so boundary counts are run. These counts cannot cover all the existing maneuvers in the network: to calculate the unknown flows, the most common procedure is the Fourness algorithm.

2.1.3. The Fourness Algorithm

The Fourness algorithm is a mathematical procedure to equilibrate the row and column totals on a square matrix.

The first step of the process is to create the “prior” matrix, that is the matrix from which the algorithm will start. First, the “fixed relations” must be defined: these are O/D relations that are completely known from the observation because there is only a possible path between them, so the flow is exactly the surveyed one. It is important, for the course of the algorithm, that some fixed relations (as much as possible) are included among the traffic surveys. Once the matrix cells defining the fixed relations are identified, there are two different construction methods, depending whether an old O/D matrix is available or a new matrix must be created from scratch. In the first case, the fixed relation cells are substituted in the corresponding cells of the available old matrix. In the second case, the empty cells (with non-fixed O/D relations) must be filled with values that represent the relative weight of each relation with respect to the others. The specific value of each cell is not important, as long as it makes evident the relative importance of each maneuver with respect to others; the correct value will be the result of the process. In general, anyway, the total generation or attraction for each zone minus the fixed relations is the value distributed among the cells on rows or columns. This distribution is an heuristic evaluation, based on qualitative observations collected during the surveys and on evaluation of the amount of trips between each

O/D pair. An important role in this process is played by the experience of the modelers. It is important to note that, because the Fourness algorithm is multiplicative, a cell with a value of 0 will remain 0 at the end of the process.

Now, with the prior matrix complete, it is possible to set up the calculations for the algorithm. The process aims to approach the column and row sums of a matrix to a set of target observed values multiplying the cells of the matrix by suitable factors, acting first on the rows and then on the columns (or vice versa). The suitable factor is the relative difference between the row and column totals at each step and the target value for each row and column. If the matrix has n rows and columns, i=1,2,…,n represents the rows and j=1,2,…,n represents the columns, the multiplicative factor Δ will be

$$\Delta_i = \frac{T_i}{R_i} \qquad \Delta_j = \frac{T_j}{R_j}$$

where Δ_i represents the relative difference on row i and Δ_j the difference on column j.

T_i and T_j represent respectively the row and column totals without the fixed relations: these are the values that the algorithm will try to equal to the reference values for rows and columns, R_i and R_j. The reference values are the generation and attraction totals for each zone calculate from observations minus the fixed relations. These values can be calculated as follows:

$$T_i = \sum_{j=1}^{n} c_{ij} - \sum_{j=1}^{n} f_{ij}$$

$$T_j = \sum_{i=1}^{n} c_{ij} - \sum_{i=1}^{n} f_{ij}$$

$$R_i = G_i - \sum_{i=1}^{n} f_{ij}$$

$$R_j = A_j - \sum_{i=1}^{n} f_{ij}$$

where c_{ij} is the generic cell, f_{ij} is the fixed relation cell, G_i is the total generation from zone i and A_j the total attraction to zone j. Once these calculations are complete, the fixed relations flows must be set to 0, because they must not change during the algorithm and must not influence the calculations; the flows of the fixed values do not need to be adjusted for they already represent reality.

Here below, an example of prior matrix ready to be used in the Fourness algorithm for a 10 zones network. The green cells are the fixed relations, already set to 0. The orange cells are the reference “target” values. The “tot” row and column represent the row and column totals. The “delta” cells are the relative difference between row/column totals and target values.

	1	2	3	4	5	6	7	8	9	10	tot	ref	delta
1	0	65	75	11	3	6	3	65	0	0	228	113	0.50
2	26	0	0	71	21	24	15	383	1	12	553	611	1.10
3	13	0	0	5	5	8	5	70	0	5	111	155	1.40
4	29	220	6	0	5	8	0	91	0	4	363	144	0.40
5	2	16	4	0	0	0	0	0	0	1	23	23	1.00
6	8	18	6	2	1	1	0	45	0	0	81	30	0.37
7	29	22	3	3	1	2	0	24	0	1	85	228	2.68
8	33	443	11	28	0	26	12	0	0	10	563	488	0.87
9	1	2	1	0	1	1	1	2	0	0	9	9	1.00
10	0	9	16	1	0	0	0	7	0	0	33	54	1.64
tot	141	795	122	121	37	76	36	687	1	33	2049		
ref	103	494	110	106	78	34	36	798	0	54			
delta	0.73	0.62	0.90	0.88	2.11	0.45	1.00	1.16	1.00	1.64			

Figure 2: example of prior matrix

The algorithm is now iterative and equals the row and column totals to their reference values multiplying each cell for the relevant Δ factor according to the formulas

for rows

$$c_{ij}^{n+1} = c_{ij}^{n} * \Delta_i^n$$

the following step is applied to columns

$$c_{ij}^{n+2} = c_{ij}^{n+1} * \Delta_j^{n+1}$$

The calculations are carried alternatively on rows and columns: starting from the results of step n, rows are equilibrated at step n+1 and columns at n+2, then rows at n+3, columns at n+4, and so on. Once the difference between factors (for rows and columns) does not change significantly and the differences between total and target values are below 3-5%, the process is considered complete. An average between the last step of row equalization (where all the Δ_i = 1) and the last step of column equalization (where all the

$\Delta_i = 1$) can be done to equally distribute the differences between rows and columns, origins and destinations. Also, each single difference from the reference value will be halved.

The following images show the final matrices for the Fourness algorithm started with the matrix in figure 1. The algorithm converged after 14 iterations (7 for rows and 7 for columns) with a difference from targets of 1%. The first matrix is the result of the last iteration for rows with all row factors Δ equal to 1, the second is the last iteration for columns with the column factors Δ equal to 1. The third matrix is the average of the previous two, and it is possible to see how the differences are halved.

R7		1	2	3	4	5	6	7	8	9	10	tot	ref	delta
	1	0	18	42	5	4	1	1	43	0	0	113	113	1.00
	2	12	0	0	55	45	9	13	460	1	16	611	611	1.00
	3	8	0	0	5	14	4	6	110	0	9	155	155	1.00
	4	8	62	3	0	6	2	0	60	0	3	144	144	1.00
	5	1	13	6	0	0	0	0	0	0	2	23	23	1.00
	6	1	4	2	1	1	0	0	21	0	0	30	30	1.00
	7	49	41	11	8	8	3	0	104	0	5	228	228	1.00
	8	25	359	18	34	0	16	16	0	0	21	488	488	1.00
	9	1	1	1	0	2	0	1	3	0	0	9	9	1.00
	10	0	8	29	1	0	0	0	15	0	0	54	54	1.00
tot		105	506	113	108	80	35	37	815	1	55	1855		
ref		103	494	110	106	78	34	36	798	0	54			
delta		0.98	0.98	0.98	0.98	0.98	0.98	0.98	0.98	1.00	0.98			

Figure 3: the last iteration of the Fourness algorithm, row equilibrium

C7	1	2	3	4	5	6	7	8	9	10	tot	ref	delta
1	0	18	41	5	3	1	1	42	0	0	110	113	1.02
2	12	0	0	53	44	9	13	451	1	16	599	611	1.02
3	8	0	0	5	14	4	5	108	0	8	152	155	1.02
4	7	61	3	0	6	2	0	59	0	3	141	144	1.02
5	1	13	6	0	0	0	0	0	0	2	22	23	1.03
6	1	4	2	1	1	0	0	21	0	0	29	30	1.02
7	48	40	11	8	8	3	0	101	0	5	223	228	1.02
8	24	350	17	33	0	15	16	0	0	20	476	488	1.03
9	1	1	1	0	2	0	1	3	0	0	9	9	1.02
10	0	8	29	1	0	0	0	15	0	0	53	54	1.02
tot	103	494	110	106	78	34	36	798	1	54	1814		
ref	103	494	110	106	78	34	36	798	0	54			
delta	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00			

Figure 4: the last iteration of the Fourness algorithm, column equilibrium

MEDIA	1	2	3	4	5	6	7	8	9	10	tot	ref	delta
1	0	18	41	5	3	1	1	42	0	0	112	113	1.01
2	12	0	0	54	45	9	13	455	1	16	605	611	1.01
3	8	0	0	5	14	4	6	109	0	9	153	155	1.01
4	7	62	3	0	6	2	0	60	0	3	142	144	1.01
5	1	13	6	0	0	0	0	0	0	2	23	23	1.01
6	1	4	2	1	1	0	0	21	0	0	30	30	1.01
7	49	40	11	8	8	3	0	102	0	5	226	228	1.01
8	24	355	17	34	0	15	16	0	0	21	482	488	1.01
9	1	1	1	0	2	0	1	3	0	0	9	9	1.01
10	0	8	29	1	0	0	0	15	0	0	53	54	1.01
tot	104	500	111	107	79	34	36	806	1	55	1835		
ref	103	494	110	106	78	34	36	798	0	54			
delta	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	1.00	0.99			

Figure 5: average of the last iterations of the Fourness algorithm

2.2. *Modeling the Demand*

Micro simulation gives the opportunity to model the demand on many levels. Road users can be divided by vehicle type, travel purpose and time of the travel. Each drivers is characterized as well according to a set of characteristics and statistical distributions of these characteristics in the population, as will be discussed in chapter 2.6.1. Here I will focus on the more general divisions of the simulated population in vehicle types, purpose for travelling and time of travel. Each of these characteristics generate a segmentation of the total O/D matrix into different layers.

Micro simulations can represent very long time periods: the transport system characteristics are likely to change in time, the most common variations being the demand variations between peak and inter peak, but also changes in lane restrictions (some bus lanes are closed only during peaks) or traffic signal timing. To take into account these modifications, the simulation period can be divided into time periods. Each time period makes editable all the simulation characteristics with respect to previous and following periods. Each time period can have a different duration. When setting different periods, demand data (matrices) are divided in different layers, each one representing the total demand present in that period. This total demand is then split into the unit time interval of simulation (in general 5 minutes) through statistical distributions (profiles) representing the real release profile of the network zones. Each O/D pair in the matrix can have a different release profile, and within each O/D pair, each vehicle type (they will all have a specific matrix), that is, each matrix layer, can have a different release profile.

So, within each time interval the demand can be divided in layers not only according to time, but also according to vehicle type. Each vehicle type can have its own demand matrix or it is possible to define for each demand matrix a proportion of each vehicle type. The most common choice is to create two matrices, one for light vehicles and one for heavy, and then assign a proportion to this matrices to simulate cars and light goods vehicles for the light vehicle matrix and to simulate medium and heavy goods vehicles for the heavy vehicle matrix. Public transport vehicles are not included in these matrices: public transport is modeled through fixed routes and timetables (in terms of frequency and time of release), both assigned to specific vehicle types.

The last characterization of the demand considers the purpose behind the travel. This issue is halfway between the vehicle and demand characterization. According to the purpose of travel, commuting, business, study, leisure, etc., it is possible to change the parameters of the cost equation. For each travel purpose, a vehicle class must be defined, so, for example, on a model there may be 3 o 4 car vehicle classes, each group with a different purpose for travelling and a different cost equation. The different cost equation is assigned to the vehicle class.

2.3. Modeling the Infrastructure Supply

The detail in the representation of the road network is one of the main differences between micro-simulation and the other transport engineering models. Each road feature can be modeled and modified, the number of lanes, the type of intersection (signalized, non-signalized, ramps, roundabouts), lane restrictions, bus or public transport lanes, etc.

The basic elements of an infrastructure network in a micro simulation are the usual nodes and links. Nodes represent intersections or any change in the road layout (number of lanes, curves, changes in slope, restrictions, etc). Links are the connections (the stretches of road) between nodes. There is anyway a different approach, that eliminates the nodes: networks are represented only by a set of links and the interchanges between these links can only happen if there is a connector between the relevant links, while with a node, by default, all the maneuvers allowed by the combinations of allowed travel directions on each arm of the intersections are permitted. In this second approach, the actual geometry of the road is in general more important, in order to define the behavior (especially in terms of overtaking and queuing) of the users.

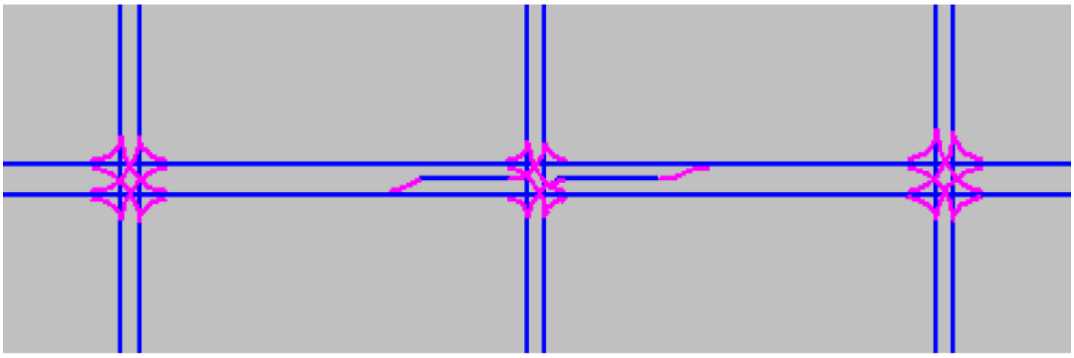


Figure 6: network with links (blue) and connectors (magenta)

2.3.1. Intersections

Any kind of intersection we find on the road can be represented: non-signalized, signalized, roundabouts and ramps. Ramps are a particular case needed for motorway scenarios. They are very similar, in terms of intersection, to a priority junction, and that is often the way they are modeled, especially in urban scenarios where the behavior of drivers is different than the one they would have on open roads. Ramps can better model the priority and merging behavior of motorways.

Common to all the intersection types is the need to define a position where the vehicles will stop (or slow down) to assess the conditions of the intersection, whether there is or not enough space or time for them to complete their maneuver. This position is defined as a stop line and represents a very important reference point for the kinematic of vehicles, as will be discussed in Chapter 2.6.4. Every link has stop lines at each end: they represent mandatory points through which the vehicles must pass and the starting point of the link with respect to intersection areas, from which the vehicles can start their desired maneuvers on links (overtaking and lane changing).

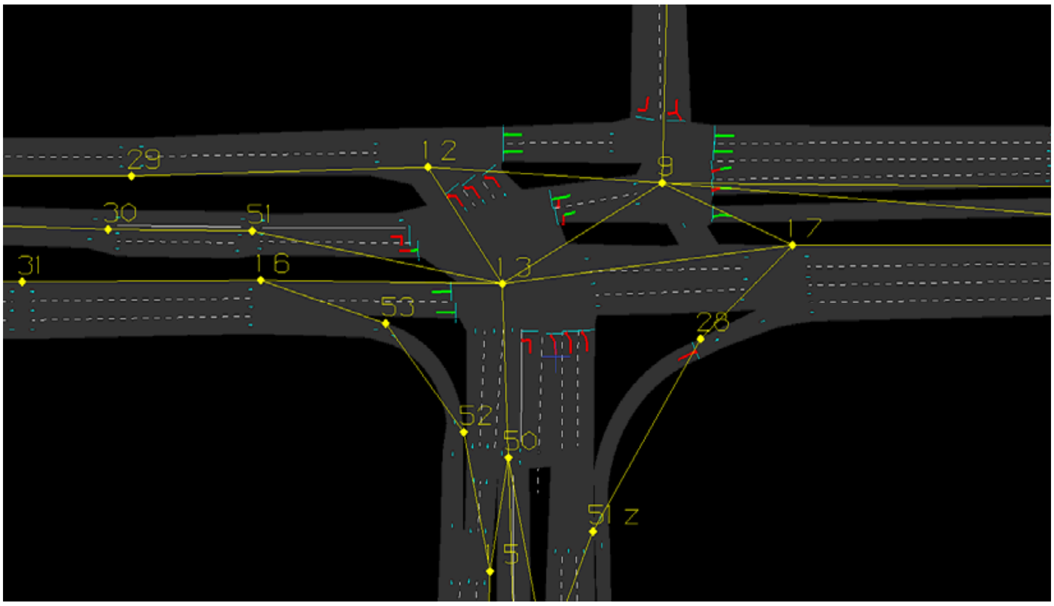


Figure 7: network with links and nodes. The blue lines and arrows are the stop lines respectively at the end and start of each link

Priority intersections are the easiest to model. If a node is present, beside the stop lines, the only aspect to define is which maneuvers have the right of way, and whether the priority is a give way (i.e. the vehicle does not come to a complete stop if there are no vehicles coming on the major road) or a stop (the vehicle comes to a complete stop every time it approaches the intersection, no matter the condition on the other road). If there is not a node, the priority is a property of the stop line on the secondary road, while on the main road there is a “priority section” informing the users on the main road that they have the right of way at the intersection.

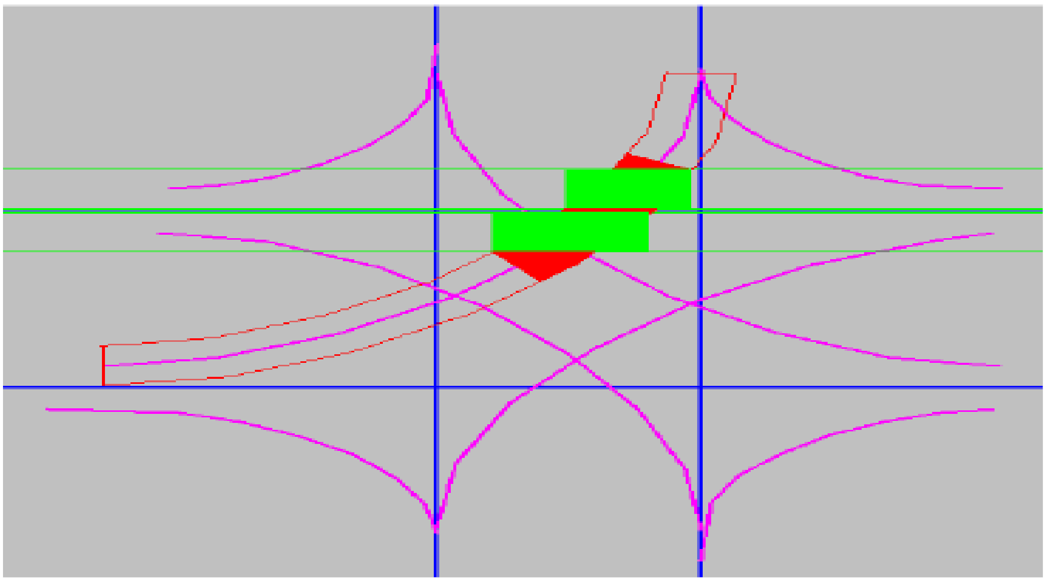
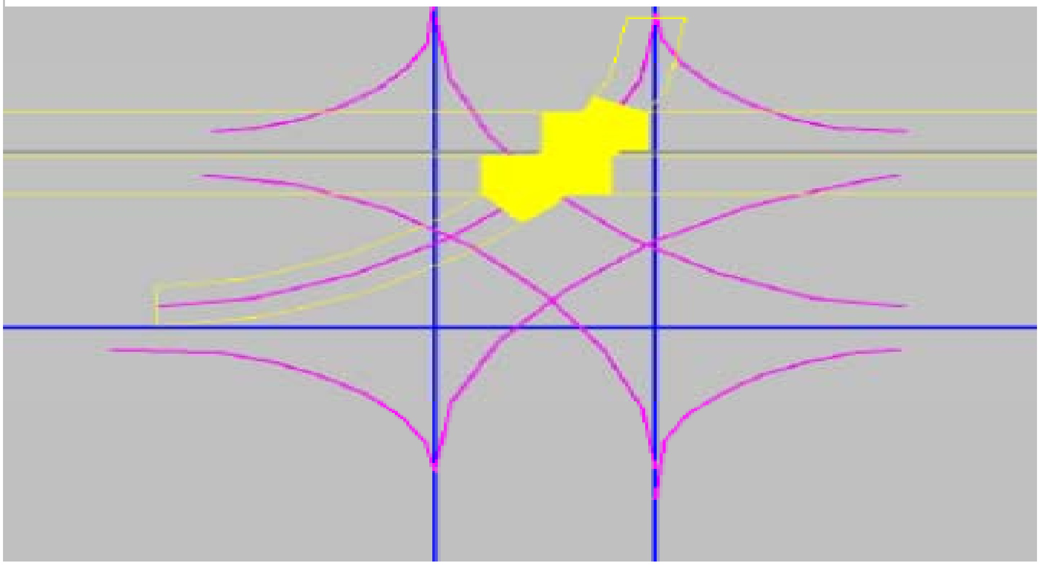


Figure 8: priority section on an intersection

Conflict areas are another possible representation of intersections: in this case, there is no need for connectors or priority or stop sections. The software will automatically calculate the acceleration and deceleration profiles, the possible maneuvers and the priorities, according to road characteristics. Also, the conflict area will create a zone in which vehicles will try not to stop, as it is possible to see in reality when drivers do not want to stop in the middle of an intersection and, if traffic is clogged on the other side, may decide not to cross the intersection.

Figure 9: conflict area; the boundaries are defined from the actual width of the lane for each maneuver



For signalized intersections the stop lines represent the position where vehicles will stop at red. Again, the presence or the absence of the node changes the way the intersection is modeled. If the node is absent it is possible to work on single lanes of the link: each lane has a traffic signal associated and the cycle definition of all of the traffic signals in the intersection will result in the behavior of the intersection, in terms of which maneuvers move together and which will have to wait. Moreover, the allowed maneuvers are defined by the connections.

If the node is present, it is not possible to work on single lanes, because the intersection is given by the node and the number of lanes is an input data from the links connected to that node. Because of this, there are three steps to model a traffic signal in this situation: first, the allowed maneuvers must be defined, together with their associated lanes (i.e., on which lanes are allowed which maneuvers); second, stages must be defined: a stage is a group of maneuvers that start during the same phase. Eventually, green, amber and all red times must be input in the model, with all the necessary offsets among maneuvers of the same phase (an offset is, for example, when the through movements stop a few seconds before the end of the phase to allow for left turns).

Vehicle actuated traffic signals can be modeled as well. Detectors must be placed on links, to count for passing vehicles. These Detector element can be modeled as simple loops or as complex areas, to model, for instance, camera detectors. Then, the signal timing plan must be created defining the average cycle, the minimum and maximum green times and the green time increase for each counted vehicle. The plan can vary on each time period.

The roundabouts are modeled as a sequence of priority nodes. If the nodes are present, the single node representing the intersection can be split in a group of nodes, each representing an access arm to the roundabout. Every single node is a priority junction. If the nodes are not present, a system of stop and priority sections must be set up, to model all the possible interactions between the lanes, both for the vehicles coming from the roundabouts and for the vehicles trying to access the intersections. Different priority rules, on the same sections, apply to different classes of vehicles according to their dimension and speed.

2.3.2. Links

Micro-simulation offers a great control over a wide range of links characteristics. This allows the user to model in detail the network layout, thus influencing drivers' behavior with high precision.

First, each link must belong to a road category: this classification influences the behavior of drivers (it is a common experience that driving on a motorway is different than driving on an urban road) and the maximum speed allowed on the link. Also, the classification of links between major or minor modifies the cost perception of users: unfamiliar drivers will have a penalty applied to the minor links to simulate their preference for major roads due to a lack of network knowledge. It is a common experience: when travelling in an unknown city, we prefer to follow the major roads because of, generally speaking, they have better signposting. Familiar drivers do not have this penalty. Penalties can also be applied to single links through a specific cost factor (with default value 1 increasable by the user): these factors are in general used to better calibrate the network. Tolls have a specific command window that adds the cost to the generalized cost function of the link.

Other basic parameters are road width and number of lanes. These two parameters have different relevance according to the software approach: some software packages consider the effective road space occupation of vehicles, so a motorbike can overtake cars even on a narrow road, even if the link is modeled as a single lane. In this case, road width is more important than the number of lanes. Lanes must be anyway modeled to reflect the reality of the infrastructure to achieve a realistic simulation, together with the space occupation characteristics of vehicles. A different approach allows only one vehicle per lane, no matter the width of the road. In this case, to simulate a wide lane, a model with two lanes to allow overtaking may be necessary. In this case the road width is little more than an aesthetic detail (the road width is mostly governed, in this case, by the number of lanes).

There are other geometric and traffic rules parameters that can be set. Road may enlarge or shrink according to downstream or upstream road layout; it is possible to set one way links or arcs (curves); flow merging and crossing maneuvers (for left turns against a large flow of traffic) can allow vehicles to force their way through opposite slow moving flows, overriding the usual priority rules. Different rules than the default ones can be applied to overtaking (on a two way – two lanes road, overtaking occupying the opposite direction is not a default behavior of micro-simulators and must be specified if it possible on that link). Also, the look for an acceptable gap for crossing a flow can be extended to the two downstream opposing links and not only to the first one. Links can be closed and each lane can be restricted to some vehicles, according to their class, height, weight or width. Bus only lanes are an example of restrictions. Restrictions can apply to all the simulation periods or just to one or some of them.

There are as well some parametric modifiers that change the default values of the simulation on a particular link for one particular time period or during the whole simulation: release and arrival rate of vehicles, visibility (this is an important parameter to model the behavior on minor links of a non-signalized intersection or ramp or roundabout), headway, target speed at the end of the link, slip lanes and toll cost. The gap acceptance for lane merge and cross and patch can be modified for the specified link. Finally, the toll cost can be set here and added to the generalized cost equation of that particular link. As usual, the changes can be permanent during the simulation or belong only to one or more periods.

2.3.3. *Intelligent Transport System*

Intelligent Transport Systems can be modeled in micro-simulation. The ITS systems are all the real time traffic information drivers can have while on the road, through radio broadcast or Variable Message Signs (VMS). GPS navigation systems can also be modeled.

In the case of VMS vehicles receive instructions as they pass a road sign. Information can be delivered to drivers as they enter a specific link or as they enter an area, defined by a list of links. This second feature models the traffic radio broadcasts. The information can be sent to all vehicles or to a single group of vehicles defined by vehicle type, aggression/awareness, destination, etc. The set of vehicle that receive an information can be also random or defined by a percentage of total vehicles.

ITS deliver information on speed or lane restrictions, delays, diversions or car park availability. These information will update the dynamic cost calculation of the simulation. ITS can also inform drivers on kinematic parameters modifications, such as modifications in aggression, awareness or target headway. This allows to modify the behavior of vehicles, or groups of them, in particular areas, such as ramps, without modifying the parameters of the whole simulation.

The simulation of navigation systems allows vehicles to change their itineraries while they are already travelling the network. In fact, the route assignment works only for not yet released vehicles. The modeling of these systems is based on time intervals, after which a new route calculation is carried using the generalized cost of the current simulation. An offset can be introduced to model the delay of navigation systems, i.e. the time between the traffic information measurement and the moment this information is delivered to the user. This offset forces the simulator to use previous data to calculate the new route.

2.4. Assignment

Assignment is the core of every transport simulation. This is the step when the network and the demand, modeled the way discussed so far, are joined to form the transport system. Considering the two approaches used to input information in the model (one based on itineraries and the other based on O/D matrices), the one based on itineraries does not require the assignment, because all the route choices are already defined by the user assigning flows to links and itineraries. The O/D matrices approach, instead, requires the assignment step to distribute the traffic through all the possible itineraries between O/D pairs.

Traffic assignment is a Discrete Choice Model theory application. Much of the mathematical theory behind the transport assignment models comes from the Discrete Choice Theory. First, a set of possible routes must be defined; then these alternatives must be analyzed and must be described the way drivers choose among the alternatives according to the previous evaluation.

Classical macro-simulation models use Static Assignment. This model considers demand and infrastructure supply are constant in time, while it is well obvious that in real life they change in time (for example, signal cycles change at different times of the day). If these changes are considered, the assignment is Dynamic and this is the way assignment is run in a micro-simulation.

2.4.1. Generalized Cost Equation

The evaluation of alternative routes passes through the definition of a travel cost for each path. This travel cost should take into account all the cost factors that drivers take into account. Because it is impossible to consider them all, the Generalized Cost Equations (the cost is defined “generalized” because it translates into monetary costs factors different from economic value, such as time and distance) use time, distance and a comprehensive factor for all other costs. The biggest part of this third factor is given by the toll, if present. The general GCE (Generalized Cost Equation) is

$$\mathrm{C} = \alpha * \mathrm{T} + \beta * \mathrm{D} + \gamma * \mathrm{P}$$

where T is the travel time, D is the length link and P represents the other costs or the toll if present. Measure units may vary, the most common being T expressed in minutes and D in kilometers. α, β and γ are parameters editable by the user. They can change according to vehicle type and/or user class (student, commuter, businessman, leisure trips, etc.). As mentioned before, a cost correction factor may be applied for particular links.

2.4.2. Route Choice

This very simple cost model, anyway, does not correctly represents reality, because people are not completely rational in their decisions, so they may perceive costs different than the effective ones and choose different itineraries. The demand will thus distribute on all the possible itineraries, somehow proportionally to their cost: the majority of drivers will use the best path, but many will use different itineraries and the combination of these changes may result in significant traffic phenomena. It does not exist an algorithm to calculate simultaneously all the possible itineraries, their cost and the relative utility they have for the users. To overcome this problem, at each iteration the best route in terms of costs is calculated. At the first iteration, with an empty network, the best route will be the minimum cost one according to the link costs defined through the network construction parameters. Then, the following iterations, dealing with the increasing traffic and congestion of the network, will calculate different minimum cost itineraries for each O/D pairs.

The cost of each itinerary is defined as the sum of all the GCs of the links belonging to it:

$$C_R = \sum_{a \in R} C_a$$

where C is the generalized cost, R is the itinerary and “a” a link belonging to the itinerary R. This cost can be used to calculate the utility of each route; the utility is the reciprocal of the generalized cost:

$$U_j = \frac{1}{C_j}$$

where U_j is the utility of route j and C_j its GC.

Obviously, the utility for a user of a route is inversely proportional to its cost. The higher the cost, the smaller the number of user that will choose the route.

In transport engineering, the most common choice function among alternatives, each one with its own utility, is the Logit function:

$$p(R_j) = \frac{e^{\mu U_j}}{\sum_i e^{\mu U_i}}$$

where U_j is the utility of route j, $p(R_j)$ the probability of choice of route j and μ (>0) the sensitivity factor of the model to utility. An high sensitivity factor will force all the users to choose the minimum cost route, while a low factor will distribute equally the users among the different itineraries.

The problem with this function is that it considers only the absolute value of the difference between utilities, so the difference between a travel time of 5 minutes and a travel time of 10 minutes is considered equal to the difference between a travel time of 105 minutes and a travel time of 110 minutes. Of course this is wrong, because in the second case the travel times are considered equal by users, while in the first case travel time doubles. A solution to this problem is given by the Kirchoff formula:

$$p(R_j) = \frac{U_j^k}{\sum_i U_i^k}$$

where k is the sensitivity of the model and the other symbols have the same meaning as before. Finally, the Kirchoff formula can be expressed as a Logit function if the GC is expressed as a logarithm:

$$p(R_j) = \frac{U_j^k}{\sum_i U_i^k} = \frac{e^{k * logU_j}}{\sum_i e^{k * logU_j}} = \frac{e^{-k*logC_j}}{\sum_i e^{-k*logC_i}}$$

where C_j is the generalized cost of route j.

To sum up, itineraries are evaluated through a generalized cost. The model calculates the GC of the minimum cost route for each O/D pair at each iteration. With the progress of the simulation and the increasing traffic in the network, the minimum cost itinerary will change due to congestion and delay. Then, a Logit model associates to every calculated itinerary during the whole simulation (one optimum itinerary for each iteration) a probability of choice and distributes the demand on itineraries according to this probability.

This is the general theory. There are, anyway, four different assignment algorithms, that can be used to speed up the simulation or to obtain more precise results. Some of them consider only the GC, some introduce a random variation in the costs and the others consider also the feedback on traffic situation from previous iterations.

2.4.3. All Or Nothing Assignment

This is the simplest assignment mode: All Or Nothing assignment considers only the GC calculated at the beginning of the simulation, with an empty network. All the demand is assigned according to this costs, regardless of congestion. It may be a quicker option to run the simulation in case of corridor model or a single intersection model or in any other case with no route choice.

2.4.4. Stochastic Assignment

The Stochastic Assignment is the simplest assignment model for networks with route choice. This model applies a random variation to the cost calculated through the GCE. The variance applied to the true cost is governed by the Perturbation parameter. This parameter can be calculated in two ways: the simplest is the percentage algorithm, where the GC is randomized with an even probability of the cost lying in the percentage ±P% around the calculated cost. For example, a perturbation level of 5 will produce a variance of ±5% from the cost calculated by the GCE.

The second algorithm is the Square Root Algorithm. The perturbation is made through a Burrell technique based on this formula

$$C' = C + \{[(N-5) * P]/500\} * \sqrt{C}$$

where C is the original link time, C' is the randomized link time (expressed in minutes), N is a random number between 0 and 10 and P is the perturbation factor, an integer > 0. Therefore, if P is 100, the cost can vary by a maximum of $\pm\sqrt{C}$.

2.4.5. Dynamic Feedback Assignment

The Dynamic Feedback assignment updates the GC considering the current level of congestion and the consequent increase of costs on some links with respect to the empty network. In this case the cost is not randomized. This cost modification influences only familiar drivers, because they have a knowledge of the network and can evaluate the effect of traffic on travel.

The frequency with which the cost is updated is the Feedback Interval. The Interval should in general be way larger than the duration of relevant events in the network (such as signal cycle times) and way shorter than the simulation period. The influence on the result of iterations depends on the method used to calculate the Feedback Factor. There are two methods: an exponential approximation and a the Method of Successive Averages (MSA).

The first gives more importance to the most recent iteration. The new cost for iteration n+1 is calculated as a weighted sum of the previous iteration n-1 cost value and the current cost value. The formula is

$$V_{n+1} = a * V_n + (1-a) * V_{n-1}$$

where a is the feedback factor (default 0.5, an high value will result in greater propensity to re-routing because an higher proportion of delays is fed back into the simulation) and V is the cost of making a particular turn from a link, respectively in the following iteration n+1 (output of the equation), in the current iteration n (input) and in the previous iteration n-1 (input). When using this algorithm with the 0.5 default value, the last iteration has a weight of 50%, the iteration n-1 of 25%, the iteration n-2 of 12.5% and so on.

The MSA algorithm is based on the following equation

$$T_i^{n+1,k} = \left(1 - \frac{1}{N+n}\right) * T_i^{n-1,k} + \frac{1}{N+n} * TO_i^{n,k}$$

where N is a user defined value, k is the index of feedback interval, n is the current iteration, i the link, $TO_i^{n,k}$ the current travel time on link i, T_i is the travel time on the previous and following iteration. The MSA algorithm gives the same weight to the current and recent iterations. This way, the influence of the increasing number of iterations is reduced.

2.4.6. Stochastic Dynamic Assignment

The Stochastic Dynamic Assignment uses perturbation in conjunction with feedback. This is the most advanced assignment model. This assignment randomizes the release times, the release link (if more than one link leave a zone), the route perturbation, and the assignment of attributes to vehicles.

This randomization is based on casual number generation. Model should be run several times, with different seeds, in order to verify the consistency and validation.

2.4.7. Micro and Macro Routing

Assignment calculations in a micro-simulation model become time consuming when the network size increase. In fact, all the previous calculations are run for each link and intersection and possible route. It is easy to understand how the number of available links, intersection and routes rapidly increase with the increase of the simulated network.

To solve this problem, it is possible to superimpose a macro network over the micro network. The macro network is more similar to macro simulation networks: nodes represent zones, car parks or waypoints and not simply vehicle release areas. Links do not represent exactly the road network but just the possible connections between nodes, regardless of their geographic distribution but taking into account the costs.

The macro level assignment considers only macro nodes and links and not every single link between them, This is a much faster way to calculate routes on large networks. Perturbation and feedback consider the whole network and vehicles are made aware of delays occurring on road section they are not yet travelling on, giving them the opportunity to re-route to different macro nodes.

At the micro-routing level the vehicles are aware of what happens only between two macro nodes or waypoints, and will try to minimize the cost of reaching the next macro node rather than the whole travel. It is easy to see how calculation are reduced this way because of the reduction of the considered network. Perturbation is applied only to the cost between two macro nodes.

Micro-level feedback has two methods to calculate feedback: the Standard Feedback and the Aggression and Awareness Method (AggrAw Method). The first method re-routes simultaneously all the familiar vehicles as soon as the cheapest route becomes congested and the second option route becomes cheaper. The second method uses the sum of aggression and awareness of each vehicle as an indication of their propensity to re-route. A familiar driver with low levels of aggression and awareness will be less inclined to re-route than drivers with high values.

Considering a normal distribution, those vehicles with high levels of aggression and awareness will reroute to avoid small delays, those in the mid-ground will reroute for moderately high delays and the vehicles with a low level will reroute only for high delays. Generally speaking, the population on the far right of the statistical distribution, no matter which one is used, will be less likely to reroute than the population on the far left.

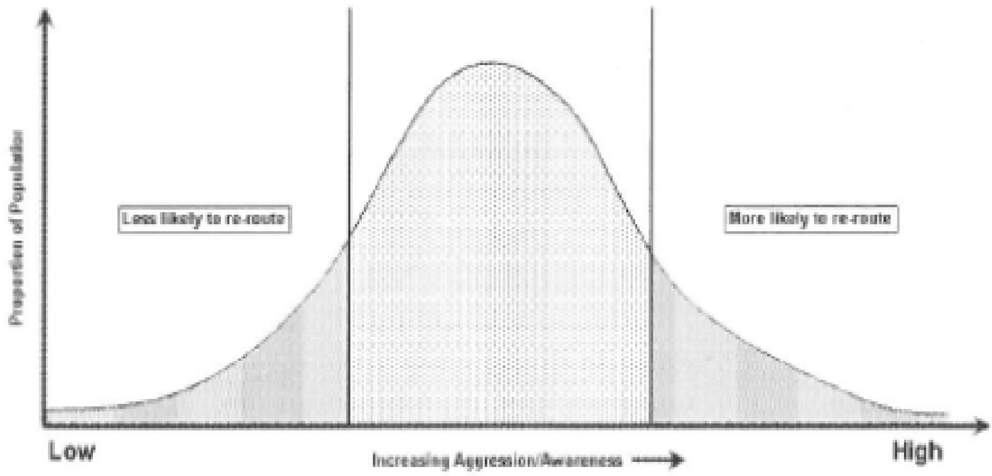


Figure 10: AggrAw method, statistical distribution of Aggression + Awareness characteristics

2.5. Microsimulation Outputs

A wide range of outputs is available from micro simulation models. Many of them are also available on macro models, but the advantage of micro simulation is that the measurement take into account effective acceleration and deceleration of vehicles and changes in the demand and road network, while macro models consider average speeds, average journey times and fixed demand and network characteristics. It is clear that the effect of speed and travel time changes is not irrelevant for emissions and fuel consumptions, especially in urban areas, but, on the contrary, emissions and fuel consumption are heavily influenced by accelerations and decelerations.

There are many statistics collection modes. Periodic sampling summarizes network statistics within time intervals. Summaries of statistics can be calculated for the whole simulation time or for specific events. Loops and paths can be defined to obtain statistics on specific links or areas.

Examples of available statistics are:

- **Pollution**: emissions level on each link according to the input emissions model; there can be a general model or specific vehicle type models

- **Turn counts**: vehicles on each link making each possible turn
- **OD counts**: vehicles on each link broken down by OD pairs
- **Releases:** counts the number of vehicles unable to be released due to congestion
- **Car parks occupancy**
- **Link delay**: the number of vehicles on each link and their mean speed
- **Bus Delay**: journey times for buses
- **Signals**: time spent by each traffic light in each phase and the number of times the period is called; this is used mainly for actuated signals
- **Paths**: journey summary for each defined path
- **Saturation flows**: vehicle flows through signals
- **Network Delay**: time spent by all the vehicles that have traversed the network
- **Trip Analysis**: journey, departure times, elapsed time and mean speed for each vehicle trips
- **Routing Paths**: routing costs for specified paths in the network
- **Incidents**: some simulators can model incidents and their effect on the traffic situation
- **Economics**: summary of network journey times and distances

Queues are a particular and very important output. The first problem is how to define when and where vehicles are in a queue. There are two parameters: speed and headway. If they fall below a threshold value, the vehicle will be considered as queued. It is possible to consider them both or consider the vehicle queued if at least one condition is verified. Another possibility is to define the minimum number of queued vehicles to consider that situation a queue. In general, the reported output for queues are average length of the queue, the maximum length of the queue and the number of stops each vehicle is forced to do, i.e. the number of times a vehicles enters in a queued state.

2.6. Specific Micro-Simulation Issues

The simulation of single vehicles arises issues that no model before had to face. If the model needs to simulate in detail the behavior of drivers in a road network, considering accelerations, overtaking, braking, lane changing, merging, etc., all those phenomena must be translated into mathematical models and equations. The mathematical theories behind most of these equations is the leader-follower model. Each vehicle (follower) is considered as if it was chasing the previous one (leader), trying to keep the headway between them to an acceptable level.

2.6.1. Driver Characterization

Even if micro-simulation considers the driver and the vehicle as one unit, the so-called Driver/Vehicle Unit (DVU), I will discuss separately the parameters related to driver behavior and the ones related to the physical behavior of the vehicle.

The behavior of drivers is essentially determined by two parameters, aggression and awareness. The first influences the gap acceptance for merging or lane changing, while the second influences the gap drivers leave to others to merge at a lane drop. Those characteristics are assigned to the population through an editable statistical distribution. The most usual one is a normal distribution. There are two more general parameters applied to the whole population, the mean headway, that is the time between two vehicles (this is modified by aggression and awareness) and the minimum gap, that is the distance left between two vehicles (fixed minimum). Each parameter can have a perturbation parameter assigned, that is an acceptable variation, in percentage, from the optimum value.

Overtaking rate is also a parameter for the population. The overtaking process involves a series of evaluations and calculations made by the model. In general, a 2 dimensional geometry for junctions and links is used to assess visibility. Effects such as blind summits are not considered, and can be modeled by barring overtaking on the involved links. To start the overtaking process, a vehicle must be restricted in speed by the vehicle ahead and it is close to it.

Then, if overtaking is permitted on the link, the potential overtaker assesses the required distance to accelerate, pass the car ahead and return to its lane. A safety margin is considered when calculating this distance. Visibility is calculated for straight links as equal to the one set in the link parameters, while for curves visibility is the chord from the front of the car to the end of the opposing carriageway is used. Across nodes, visibility is worked out from the angle of the in and out link. If this check does not find any oncoming vehicle the overtake takes place.

2.6.2. Vehicle Characterization

The vehicle parameters can be divided into three groups: general, physical and dynamics parameters.

General parameters include vehicle type (cars, light, medium, heavy goods vehicles, buses, but also divisions by trip purpose, such as car-commuting, car-leisure trips, car-business trips, etc.). Other parameters are the familiarity with the network that influences the cost perception of minor road links, perturbation of the equations, age, trasponders to receive ITS signals and emissions model.

Physical parameters include length, width, axle, kingpin, height and rear axles dimensions. It is possible to add sections to the vehicle to simulate, for example, road trains, bi-articulated buses or trams with more than two sections.

Dynamics parameters include weight, maximum speed, acceleration and deceleration rates, drag and inertia. More detailed parameters are often available for heavy goods vehicles, the most critical road users. They include the effect on deceleration and maximum speed of age, gravity, incline, power base and power divisor.

2.6.3. Generation of Vehicles

The generation of vehicles on a micro simulation deals with the modeling of a traffic flow in terms of vehicle types, speed and interactions on a link. The main parameters for vehicle generation are headway, vehicle type, initial and desired speed for vehicles.

The simplest headway model is the *Uniform Headway Model*. It is an obvious simplification of reality, assuming that all vehicles have the same headway. The uniform time headway t can therefore be calculated as

$$t = \frac{3600}{Q}$$

where Q is the unidirectional flow on the considered link in vehicles per hour.

The Shifted *Negative Exponential Headway Model* represents a random arrival of vehicles. This model is in general used for small flows situation. In fact, the model uses a Poisson distribution to model the arrival of vehicles. The assumption behind this distribution is that the probability of a vehicle arrival is independent from the arrival of any other vehicles. This may be real only for low flow situations. The probability of arrival of a vehicle during the small time period Δt is λΔt, where λ is the mean arrival rate of vehicles. If the arrival of vehicles is represented by a Poisson distribution, then the distribution of headways is a negative exponential distribution. The shift is the minimum headway value specified for the function below which the generated headway can never fall. This represent the safety distance between two vehicles. Under these assumptions, the function representing the distribution of headways is

$$f(t) = \lambda e^{-\lambda * (t-\tau)}$$

where t>τ is the headway value in seconds, τ is the minimum headway (shift value) and $\lambda = \frac{1}{\bar{t}-\tau}$ with $\bar{t}$ representing the uniform headway as calculated before. The new headway can be calculated from the following equation

$$t = \tau - (\bar{t} - \tau) * \ln(N)$$

where N is a random generated number between 0 and 1.

The *Composite Headway Model* considers that the overall headway distribution for vehicles is given by two contributions: platooned and non-platooned vehicles. The headway distribution for platooned vehicles comes from a normal distribution, while non-platooned vehicles headway are distributed according to a shifted negative exponential distribution. The proportion of non-platooned vehicles is defined as $P_{Np} = 1 - P_p$, where P_p is the proportion of platooned vehicles. The mean time headway for non-platooned vehicles is

$$\bar{t}_{Np} = \frac{\bar{t} - \bar{t}_p * P_p}{P_{Np}}$$

where $\bar{t}_p$ is the mean headway for platooned vehicles and $\bar{t}$ is the uniform headway as calculated above. If $\bar{t}$=$\bar{t}_p$ all the vehicles are platooned and $P_p = \frac{\bar{t}_p}{\bar{t}}$. It is possible to calculate the new vehicles' headways starting from the value of $\bar{t}$; if $\bar{t} < \overline{t_p}$ the new headway is chosen from the normal distribution. P_p is estimated from input $\bar{t}_p$ and P_{Np} and $\bar{t}_{Np}$ can be calculated. A random number N between 0 and 1 is generated and if $N \le P_p$ the next vehicle is considered part of the current platoon with an headway chosen from the normal distribution, otherwise the vehicle is not platooned and the headway is chosen from the shifted negative exponential distribution. This model is used for intermediate situation between free flow and congested traffic.

For denser traffic the suggested generation model is the *Lognormal Headway Model*. The probability function for this model is

$$f(h) = \frac{1}{\sigma * t * (2\pi)^{\frac{1}{2}}} * e^{\left[-\frac{1}{2\sigma^2} * (\ln(t) - \mu)^2\right]}$$

The most common algorithms are anyway the Composite Headway and the Shifted Negative Exponential.

The vehicle type is defined starting from the input fleet characteristics composition. From these data, the probability of given vehicle of being a defined vehicle is calculated and termed p(i), where i represent the generic vehicle type present in the simulation. Obviously, $\sum_{i=1}^{n} p_i = 1$. Then a random number N from 0 to 1 is generated. Then, this number is verified against the vehicle type with the minimum probability: if $N \leq p(i)_{min}$ the vehicle will belong to that class. If $N > P(i)_{min}$ the second lowest probability will be added. If N minor or equal of this new probability, the vehicle will belong to this second class. If not, a third probability will be added, and so on. As an example, for a model with three vehicle classes (cars, light goods vehicles and heavy goods vehicles) each one with its own probability (respectively p(c), p(l) and p(h), with the highest probability of being a car and the lowest of being a heavy goods vehicle), and a random generated number N, if N≤p(h) the vehicle will be heavy, if N≤p(h)+p(l) the vehicle will be a light goods vehicle and if N≤p(h)+p(l)+p(c) the vehicle will be a car. The last equation is always true, because p(h)+p(l)+p(c)=1 representing the total fleet in the model, so for each random number a vehicle is generated in the network.

The desired speed is the speed a vehicle will try to achieve on a specific section in free flow conditions. The distribution of desired speed is in general assumed as normally distributed, with a coefficient of variation calculated from observation.

The initial speed is the minimum between desired speed and calculated safe speed. The safe speed comes from the car following algorithms (that I will discuss in the next paragraph) with initial acceleration set to 0 and solved for the following vehicle speed.

2.6.4. Kinematic of Vehicles

As mentioned before, in a micro-simulation model the driver and the vehicles are considered as a unit, called DVU (Driver/Vehicle Unit). Therefore, the models controlling the kinematic of vehicles travelling in the network often involve drivers' characteristics.

The basis of drivers behavior in terms of acceleration and deceleration is the target headway: this is the gap the driver will try to achieve and maintain from the driver in front of him. According to the variation of this gap, that is, according to the variation of speed of the vehicle in front, the DVU will shift among the different travel states: acceleration, cruising, deceleration.

The mean target headway is 1 second. Because real drivers change their behavior according to a number of situations and depending on their driving characteristics, variations are often introduced to model these different situations. Here below, variation coefficients from S-Paramics software.

Condition / Parameter	Variation Factor
Vehicle type: e.g. Heavy goods vehicles	1.8
Presence of single-lane highway (no lane changing possible, as in road-works),	1.5
Weather, Lighting (fog, rain, darkness) (unvalidated, just for example)	1.1 - 3.0
Close to motorway merge (accept smaller headway for limited time)	0.5
Close to traffic signals: straight ahead Close to traffic signals: turning left	0.5 1.1
Aggressiveness (everywhere) $Ag \leq 4$ $Ag \geq 4$	 $\frac{6-Ag}{2}$ $\frac{12-Ag}{8}$
Awareness (Near Lane-Drop) $Aw \leq 4$ $Aw \geq 4$	 $\frac{Aw+4}{8}$ $Aw - 3$

Chart 1: target headway variation coefficients from S-Paramics

So, if not constrained by an approaching junction, a DVU will try to maintain this target headway from the vehicle in front varying its speed accordingly. The reaction time of these speed variations is in general 1 second. It is obtained by giving to each DVU a short memory, recording some positions and speeds in the past beside its current speed an positions. Many points would give a better accuracy, but longer calculation times. This is a typical trade-off in calculations, speed for accuracy. The reaction time is modeled by basing the acceleration calculations on the speed at which the DVU in front was travelling at some point in the past (in case of a 1 second reaction time, this point will be 1 second before current time). The reaction time is used to simulate shockwave effects in traffic flows.

From the target headway comes the target point: this is a position at a distance s behind the leading DVU calculated basing on the desired headway and the current perceived speed of the leading DVU. The current perceived speed is the actual speed at some point in the past, due to the reaction time modeling. In the scheme below, V_2 is the speed of the follower, V_1 the speed of the leader, t the target distance, h the target headway and g the current gap. h is expressed in seconds, s,t,g in meters.

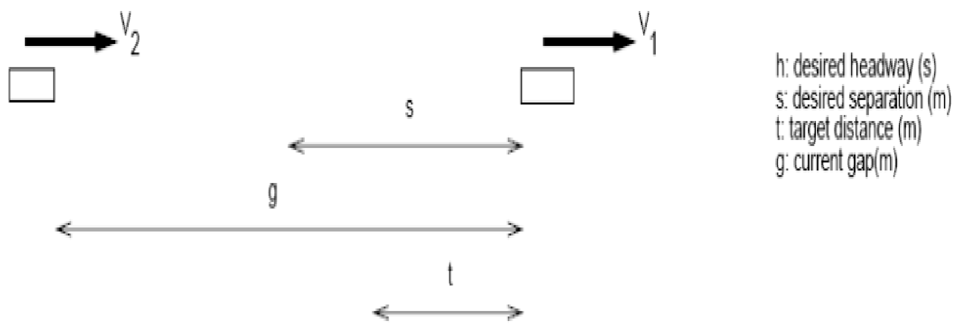


Figure 11: leader-follower model diagram

The distance s is calculated as

$$s = h * \Delta V$$

where $\Delta V = V_1 - V_2$. To achieve a more realistic platooning of the vehicles, the target point position is calculated as

$$t = \frac{s^2}{g}$$

The bunching acceleration c describes the kinematic of vehicles forming a platoon

$$c = k_1 \frac{g - s_{min}}{g}$$

where s_{min} is the minimum distance between two vehicles and k_1=1s^{-2}. In S-Paramics and Vissim, for instance, the minimum distance is 2 meters.

The relation between target headway and velocity difference can be represented as a space with 5 different regions, as shown in the figure below.

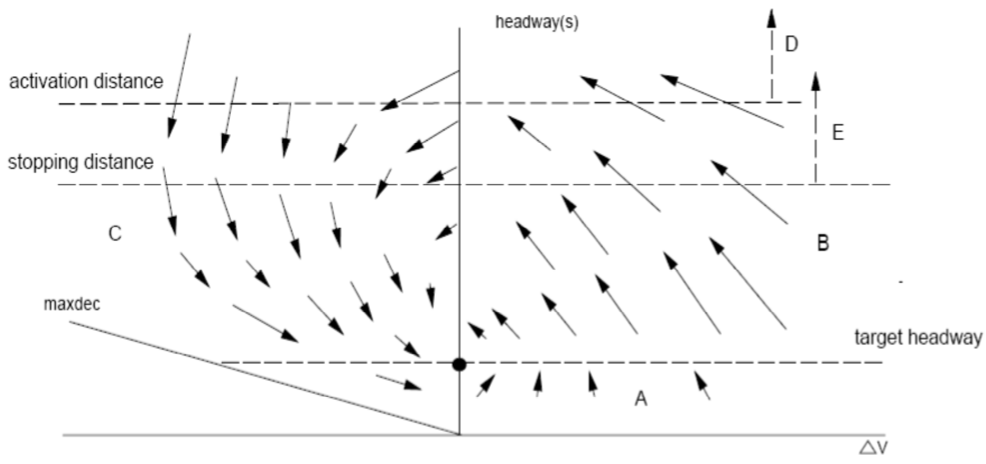


Figure 12: headway/velocity-difference space with acceleration regions

To each region, named from A to E, there is a corresponding acceleration, named as well a_A to a_E.

In phase A, the DVU has overshot the target point, that is, the headway is less than the desired one. The DVU will try to reach the desired headway as fast as he can according to its physical constraints. The acceleration is then

$$a_A = k_2 * \Delta V$$

where k_2=1s^{-1}.

If the leading DVU is pulling away from the following DVU, this last will accelerate to maintain the desired headway. This is region B in the figure and the related acceleration is

$$a_B = k_2 * \Delta V + k_1 * \frac{g - t}{t}$$

Region C represents a situation where DVUs are at constant separation or coming together

$$a_C = c - \frac{(\Delta V)^2}{g - t}$$

These were the so-called cruising modes. These mode are defined by acceleration and deceleration comprised within a certain range, differently defined in each micro-simulation model but anyway consistent with vehicle physics. If the deceleration of a vehicle overcomes a certain threshold, the following DVU perceive the leading DVU as in a braking state. When the leader is braking, its perceived speed will be decreased by an amount dependent on its maximum deceleration rate. This models the expectation of the follower that the speed of the leader at the next time step will be considerably smaller than the previous one. This will induce the follower to overcompensate the braking with an intensity inversely proportional to the distance between the vehicles. Together with the reaction time, this models the shockwaves effects of braking in a traffic flow. This is region D of the figure 8 above. However, because the speed of the DVU ahead is predicted and may be equal to 0, a test is run to check whether the following DVU is in danger of collision or not. If not, the acceleration is set to a positive value and the acceleration in D region results

$$a_D = 1 \ {}^{m}/_{s^2}$$

If the leading DVU is perceived to be accelerating at an high rate, the follower will set its acceleration to the its maximum according to the physical constraints of the vehicle. The acceleration in region E will be then

$$a_E = a_{MAX}$$

2.6.5. Merge

Another important aspect of the kinematics of a micro-simulation model is the merging, or lane changing. This is the problem of a vehicle travelling on a lane wishing to reach a target lane merging into another flow of traffic.

The leading parameter for this maneuver is the gap acceptance. The gap acceptance is related to the target headway. If the traffic in the current lane and in the target lane is travelling at constant speed, a gap must exist both in front and behind the position the DVU would occupy that is large enough for the DVU target headway. If the lanes are travelling at different speeds, this gap must take into account the time the DVU needs to accelerate or decelerate to the new speed. If DVU_0 is the vehicle looking for lane changing and DVU_1 and DVU_2 are the vehicle respectively in front and beyond the position DVU_0 would occupy in the target lane, then the lane changing maneuver will happen if and only if two conditions are verified:

$$g_1 > d_{\Delta V_1} + h * v_1 \qquad g_2 > d_{\Delta V_2} + h * v_2$$

where

$$d_{\Delta V_1} = t_0 + \frac{\Delta V_1}{D_0}$$
$$d_{\Delta V_2} = t_0 + \frac{\Delta V_2}{D_0}$$

$$\Delta V_1 = v_1 - v_0$$
$$\Delta V_2 = v_0 - v_2$$

where v_n is the speed of the DVU_n, D_n is the maximum deceleration rate (braking rate) for the DVU_n, g_1 is the gap between the back of DVU_1 and the front of DVU_0 and g_2 is the gap between the back of DVU_0 and the front of DVU_2.

2.6.6. Turns at Intersections

The modeling of vehicles on arc, both straight and curve, uses a mono-dimensional model based on distance, speed and acceleration. The situation is more complicated on intersection. To realistically simulate the narrow turns that vehicles do at intersections, it is necessary a bi-dimensional model. This increases the amount of calculation required to model the vehicle movements, but , at any time, only a small amount of vehicles, compared to the total population, will be engaged on turning maneuvers.

The model is based on a triple (x,y,bearing) that describes both the position of the point a vehicle should head to from any arm of an intersection and the required angle of orientation once it gets there. This methods is much easier and less compute-intensive than specifying centre points for turning arcs, as it requires only one vector for each lane of exit from a junction, no matter the origin point. Moreover, the curve produced by these algorithms is more realistic.

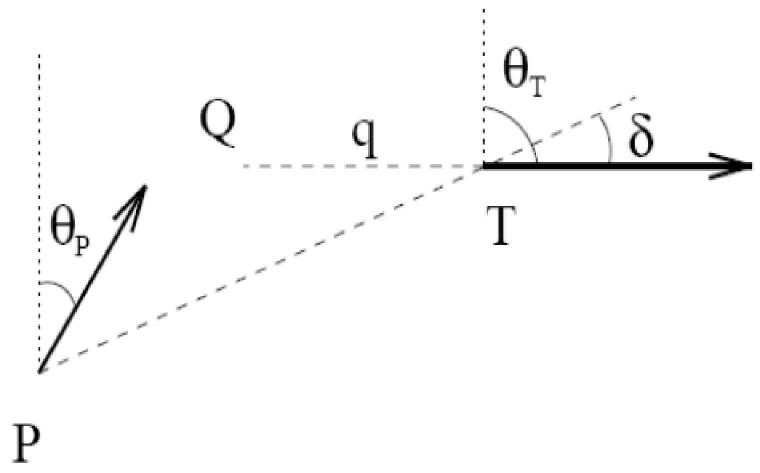


Figure 13: steering heuristic for turns less than a quarter circle

The algorithm is iterative: at each time step, the imaginary destination point is calculated and updated so that the vehicle can reach the destination point with the correct orientation.

Each step of the algorithm can be described as follows: the vehicle arrives to the position P with bearing θ_P and needs to reach point T with bearing θ_T. δ is the difference between the final bearing θ_T and the bearing the vehicle would have with a direct approach. At a distance $q = \overline{PT} * \sin\delta$, where $\overline{PT}$ represents the distance between P and T, is placed an imaginary point to which the vehicle will head. If δ is larger than a defined threshold (a typical value is $\pi/4$), the imaginary point is re-placed at a perpendicular distance r_{MIN} from the position of Q previously calculated. Once δ goes below the threshold, Q is replaced to its original position at the end of q. If the intersection lacks of space, the vehicle will head directly to the destination point with the correct bearing along the curve with the minimum possible radius.

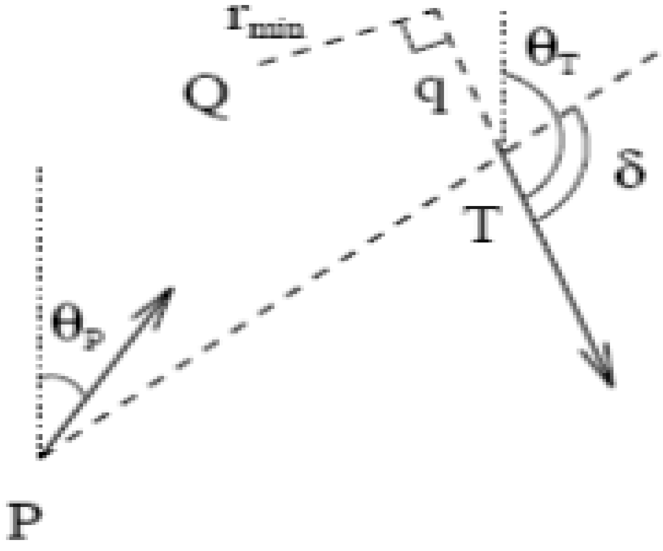


Figure 14: steering heuristic for turns more than a quarter circle

The rate of change of bearing (that is, the radius of the turning circle) has as constraints the current speed and the physical attributes of the vehicle. Given the speed, the physical constraints that determine the curve radius for a vehicle are the friction of wheels on pavement and the limits of steering mechanism (full lock).

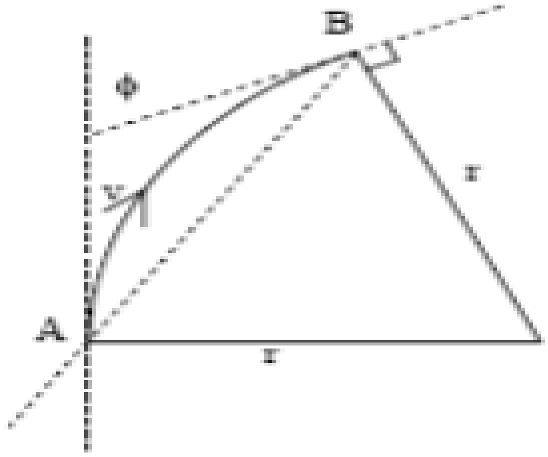


Figure 15: transition from A to B, with fixed speed and steering angle

The tire friction coefficient is f_s=0.2-0.4. The speed of turning around a corner is constrained by

$$v_{max} < \sqrt{f_s * g * r}$$

where g is the acceleration of gravity (9.81 m/s^2), f_s the friction coefficient and r the radius of the curve.

If the movement from A to B takes a time Δt travelling at the maximum speed v_{max}, the travelled distance will be v_{max}*Δt. According to these constraints, the angle of rotation of the vehicle will be

$$\varphi = arcsin\left(\frac{v_{max} * \Delta t}{r}\right)$$

Substituting for r we obtain

$$\varphi = arcsin\left(\frac{f_s * g * \Delta t}{v_{max}}\right)$$

And in the general case

$$\varphi < arcsin\left(\frac{f_s * g * \Delta t}{v}\right)$$

The second constraint on φ comes from the steering mechanism of the vehicles. If the wheels of a turning vehicle are offset from the straight position by an angle θ and the length of the wheel base of the vehicle is L, then the curve radius r will be

$$r = \frac{L}{\sin \vartheta}$$

If θ_{LIMIT} represents the maximum wheel offset (full lock) then the minimum curve radius for the vehicle will be

$$r_{MIN} = \frac{L}{\sin \theta_{LIMIT}}$$

If the vehicle is travelling its minimum turning circle at speed v, in the time interval Δt the angle turned will be

$$\varphi = arcsin\left(\frac{v * \Delta t}{r_{MIN}}\right)$$

The second constraint on the turning angle will be then

$$\varphi < arcsin\left(\frac{\Delta t * \sin\vartheta_{LIMIT}}{L} * v\right)$$

The combination of the two constraints is showed in the figure 12 below. The shaded part represents the allowable values of θ

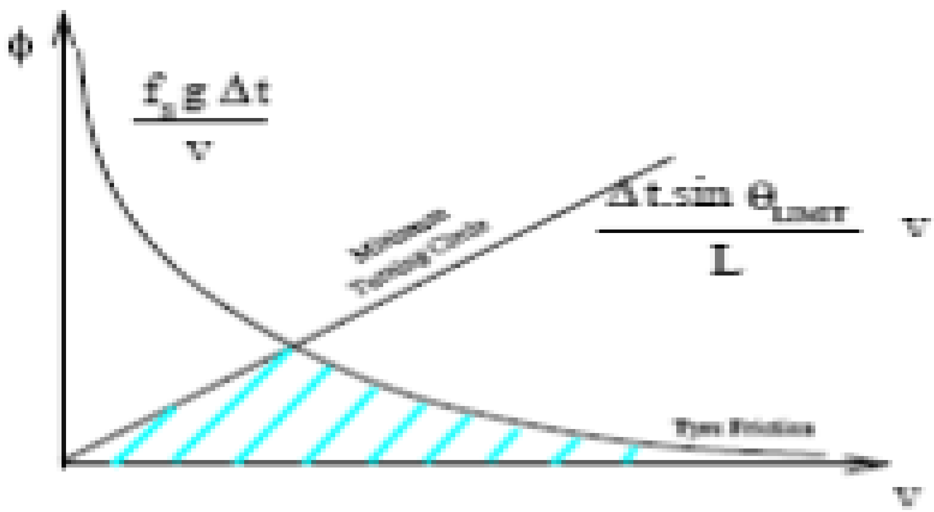


Figure 16: constraints on the steering angle of a vehicle

2.6.7. Car Parks

There are two possible representations of car parks in micro-simulation models: roadside car parks or zones. The two possibilities are used for different scopes and in different kinds of simulation. Roadside car parks are used for simulation with fixed itineraries to simulate short stops during the itinerary. Car parks behave similarly to zones in dynamic assignment simulations.

Roadside car parks can be defined as new lanes on an arc or an existing lane can be transformed in a car park. Vehicles will reach the car park using specific partial itineraries. Each car park has a capacity determined by the dimension of the stalls. Each stall will have associated an attractiveness value. Vehicles will occupy the stall with the highest value at the moment they cross the section of choice for the car park itinerary. In this representation, a vehicle will occupy a space only if it is longer than the vehicle of a certain threshold. If not, the vehicle will occupy two consecutive spaces. If the consecutive spaces are not available, the vehicle will wait until a suitable space is free (unrealistic situation). In this kind of simulation, because they have fixed itineraries, vehicles cannot re-route to another car park in case there are no spaces available. In this case, the vehicles will continue on their itinerary without stopping.

On the contrary, when car parks are used connected to zones to attract and generate vehicles in a dynamic assignment situation. Each zone can have more than one car park assigned, but each car park can be assigned to a single zone. The main difference between car parks and zones is that car parks have a maximum capacity. Once the car park is full, the vehicles will queue at the car park entrance. After waiting for a certain time, determined by the modeler, the vehicle will re-route to the next car park. If all the car parks are full, the vehicle will wait at the last car park until a place in a car park becomes available.

The “next” car park is actually the next best choice, in terms of costs, for the vehicle in relation to its destination zone. This is expressed in the form of an Utility function, similar to the ones used in the Logit choice model.

A car park utility function is defined as

$$U_{k,s} = \alpha_{k,s} * C_{parking} + \beta_{k,s} * A + \gamma_{k,s} * D_{dest} + \delta_{k,s} * D_{veh} + \varepsilon_{k,s} * fs$$

where α, β, γ, δ, ε are user-defined coefficient, C_{park} is the cost of parking, A is an attraction coefficient, D_{dest} is the distance from the destination of the vehicle, D_{veh} is the distance from the current position of the vehicle, fs is the amount of free spaces and k,s are indices related, respectively, to the vehicle type and the purpose of travel, if defined.

Each car park will have a maximum stop time, after which vehicles will leave. On the vehicle side a stop time can be defined for each vehicle type and travel purpose. If this is not defined, the default stop time is one hour. Each itinerary can also have assigned a statistical distribution of stop times according to the travel purpose to which the itinerary is associated.

The definition of the stop time allows to model parking fees: they can be hourly fees or fixed fees, in this case ignoring the stop times.

3. A CONFRONTATION AMONG MICROSIMULATION SOFTWARE PACKAGES

The previous chapter outlines the most common methodologies to create micro-simulation models. Obviously, there are more models, especially when it comes to car-following, lane changing, merging and intersection behavior algorithms.

There are commercially available many software packages, each with its own approach to the problem. The most common are S-Paramics and PTV Vissim. Open Source softwares are becoming more common, the most important so far being SUMO (Simulation of Urban MObility), created by the DLR (the German space agency) with contributions from many universities, such as Köln, Innsbruck, Lübeck, Berlin, Münich, Turin and Wroclaw.

In this chapter, I will briefly discuss the different approaches to micro-simulation of S-Paramics, PTV Vissim and Sumo, following the main issues outlined in Chapter 2.

3.1. S-Paramics

S-Paramics is a micro-simulation software from United Kingdom. It essentially a vehicular traffic simulator with a very poor capacity of simulating pedestrians, considered only as a delay for vehicles at intersections. It does not have the possibility to consider multi-modal schemes and the public transport is considered only in its interactions with the rest of the traffic flow. Thus, adding bus services on a route will not modify the modal split on that route. This is anyway, strictly speaking, out of the micro-simulation goals. While S-Paramics is essentially a standalone software, researches have been carried out for an integration with DIADEM, a public transport modeling tool, in order to simulate variable demand situations where the detailed simulation of traffic and the outputs from micro-simulation are used to update a public transport model.

The most common demand input mode in S-Paramics is the creation of O/D matrices. This is a very effective way to speed up the input phase for large models and allows for the simulation of route choice. Moreover, S-Paramics include the macro-routing option, in which a macro-network, representing waypoints and connections among them, is superimposed on the micro-network: for large networks, routes are calculated following this macro nodes and then, at micro-level, considering as route only the path from a macro node to the next one, thus reducing the necessary calculations. The union of these partial itineraries results in the complete route from Origin to Destination.

In corridor models or small models where there is no route choice, such as verification of intersections, it is possible, instead of the O/D matrix, to input directly traffic flows on specific itineraries. The creation of an O/D matrix for an intersection from site observations is anyway very fast.

The network in S-Paramics is represented through the usual node-link system. Nodes represent intersections or geometry variations in the road (number of lanes, curves, beginning of restrictions, etc.). Links are the stretches of road connecting nodes and carry all the street properties: number and use of lanes, maximum speed, cost, road category (thus defining users' behavior), etc.

The assignment models are very well developed in S-Paramics. There are four assignment models, from the All Or Nothing to a Dynamic Stochastic model that considers both perturbation (a variation of cost perception among different users) and feedback from the current iteration of the model (congestion and delays). If the flow itineraries are specified as input, there is no assignment and the model runs faster.

The outputs of the simulation are the ones outlined in Chapter 2.5: general statistics on the network, such as number of simulated vehicles, mean speed, delay and travel time. Information are available separately for public transport vehicles. Traffic counts on links or single turns are available. Queues as well have specific outputs, such as maximum and mean length or number of vehicles or the number of times a vehicle is forced to enter in a queued state (i.e., the number of times a vehicle stops). Outputs can be obtained for the whole network, for single paths of for areas of the network through the definition of a loop.

In a micro-simulation software, the driver and the vehicle are represented as a unit, called DVU (Driver/Vehicle Unit). S-Paramics models this unit according to a series of parameters editable by the user. The main driver behavior parameters are Aggression and Awareness. These are represented through editable statistical distributions, with a normal distribution as default. The target headway is the separation from a vehicle to the one in front of it and the DVU will accelerate or decelerate to maintain this headway. The speed will change according to the recorded speed of the leader at some point in the past (1 second) to simulate the reaction time. Vehicles have many different parameters to determine their dimensions, speed, acceleration, maximum deceleration, weight, etc. All of them are editable. A more detailed representation can be done for heavy good vehicles, taking into account, for example, age, because they are the most critical elements in traffic.

The kinematic model used in S-Paramics follows the one detailed in Chapter 2.

The main weakness of this software is the modeling of car parks: while it works correctly for off-road parking areas, it cannot simulate roadside car parks except as incidents: to model roadside parking, an Incident event must be assigned to a certain percentage of vehicles on a link, with a suitable delay to simulate search, maneuvers and stop times.

3.2. Vissim

VISSIM is a micro-simulation software from Germany produced by PTV. With respect to S-Paramics, VISSIM considers a wider range of road users, including pedestrians and cyclists. So, VISSIM can be used to simulate pedestrian areas and zones with traffic restrictions considering all the road users and their interactions. VISSIM can import data from other software, such as networks from VISUM (macro-simulator from the same software house) or any other traffic simulator using .ANM network files, or data from Sitraffic Office or from SYNCHRO for traffic light optimization. VISSIM can as well export data suitable as VISUM inputs. As with S-Paramics, it is not possible to simulate multi-modal processes or modal shifts. VISSIM considers the effective dimension of vehicle in terms of road occupancy when calculating maneuvers, turns, overtaking, etc., while S-Paramics considers that only on a single lane there cannot be two vehicles side by side, no matter the dimension of the road and the vehicles.

The main input mode for VISSIM is the itinerary approach. Consequently, VISSIM cannot simulate route changes due to changes in the infrastructure system. It can verify the effect on a flow of changes in traffic light, for instance, but it cannot calculate the changes in route choice due to this modification. VISSIM includes a dynamic assignment option based on O/D matrices, but it is not as good as S-Paramics in assignment and it is not often used. A common reasearch application of VISSIM is its use in conjunction with meso-simulators such as MEZZO. In Stockholm, while the city-wide network was created with the meso-simulator, specific issues have been studied with VISSIM models connected to the larger meso-model so that the effects of design changes influenced the meso-model, changing the wide area traffic distribution. This new distribution is then used as input for VISSIM in an iterative optimization process.

The fact the software is mainly designed to have fixed itineraries within a network, specializes VISSIM mainly as a verification tool for very local issues, such as intersection optimization or other situation with no route choice. It is very difficult to use VISSIM to simulate large networks. The macro and micro routing options do not exist in VISSIM.

The main outputs from VISSIM are similar to the ones in the other micro-simulation software packages.

Anyway, VISSIM has probably more detailed outputs related also to a single vehicle regarding emissions, speed, acceleration, queue times, power, position with respect to other vehicles, etc. Also the public transport outputs give a wider range of data: delay of services, boarded and alighted passenger at each stop, waiting time and service time for passenger, and average waiting time at stops.

The driver behavior in VISSIM uses as fundamental parameter the distance between vehicles. The software includes two car-following models: the Wiedemann 74 for urban areas and the Wiedemann 99 for motorways. These are also called psychophysics perception models. The first depend on the mean stopping distance and on the safety distance to avoid collisions. The Wiedelmann 99 model depends also on headway, speed difference and target acceleration from still position and from 80 km/h. Queuing behavior depends on the distance at which the DVU perceive the presence of the queue and from the number of vehicle ahead of it that the DVU controls to decide its behavior. The reaction time is defined as “sleep” parameter: it models the lack of attention of some drivers giving the user the possibility to edit the duration of reaction time and the percentage of drivers that will have a lower attention. The lane changing maneuver calculation algorithm depend on user defined maximum acceleration and deceleration parameters as well as on safety distance and on the deceleration needed to allow vehicles from a different lane to reach their target lane. Acceleration and deceleration parameters can be defined for the current vehicle and for the following one. Working on the lateral distances between vehicles is also needed to simulate the overtaking of small vehicles in the case of wide lanes. The distance parameters influence also the saturation level of the links: obviously, if vehicle have smaller acceptable safety distances, the saturation level, that is the number of vehicles that can travel on the road in free flow conditions, increases. Other relevant parameters are speed and number of heavy vehicles.

The characteristics of vehicles are modeled through editable statistical distribution: speed, weight, power, stop times at parking lots, intersections or public transport stops and the parameters for vehicle emissions are represented by statistical distributions that the user can edit through control points. These parameters are different for each vehicle class (defined as a combination of vehicle type and travel purpose).

In the end, car parks in VISSIM are better modeled than in S-Paramics. It is possible to model roadside car parks defining the stop time and the most attractive spots. Also off-road car parks present a better behavior than the ones in S-Paramics, especially in terms of occupancy, queuing and re-routing.

On the biggest problem in VISSIM is that after waiting 60 seconds to complete the desired maneuver, the vehicle that cannot complete the operation disappears from the model, because such a long waiting time is deemed unrealistic in real life. A more realistic solution would be the vehicle travelling around in loops until an opportunity to complete the maneuver appears.

3.3. *Sumo*

SUMO (Simulation of Urban MObility) is an open source traffic simulator created by DLR (the German space agency) together with a group of universities. Being open source, it is still under development and the number of available features keeps increasing. It is the most important open source traffic simulator, at least in Europe, being used in a number of researches on traffic lights evaluation, route choice and re-routing exercises, evaluation of traffic surveillance methods, simulation of vehicular communication and traffic forecast.

This last use outlines the biggest difference between SUMO and the other software considered: SUMO is capable of dealing with large networks until up to 10.000 links. The name of the software expresses this difference: SUMO is a multimodal model that can be used to evaluate the effect on modal shift and traffic composition in changes in the transportation infrastructure and policy. In SUMO, if I add public transport opportunities, the number of cars in the network will decrease accordingly to the number of people changing their travel mode.

This ability depends on the way the demand is input in the model: SUMO comprises both the models described before (fixed itineraries, generation from loops on the road and O/D matrices), but also an activity generator for the population, given their characteristics as described in census or available statistics. Starting from these data, SUMO is able to generate the demand for a city and calculate the modal shift between public transport, walking and private cars. The amount of required data is quite extensive:

on the infrastructure side, beside all the usual inputs on road and intersection layouts, the public transport network layout with bus station and the city gates (the roads from which commuters reach the city) with a proportion of inbound and outbound trips are required .

On the demand side, for generation purposes the required data comprise: number of inhabitants and households, age of residents (children, adults, retired), car ownership, maximum walking distance, unemployment rate and number of commuters entering or leaving the city. Parameters for the population are probability of car preference, estimation of time needed to drive 1km, proportion of random traffic in the whole traffic demand, variance of departure times, age distribution (what is of interest is the age interval of active population).

On the attraction side, opening and closing hours for all city's type of work position and the proportion of activities with each defined work hours, the density of residents and work places in each street. Schools with position, age interval, capacity, opening and closing hours must be defined.

Once the statistics data are complete, the Activity Generator divides travel purposes between two groups: Work and School (children) and Free Time . Free Time activities are divided into day activities (for retired and unemployed), evening activities (for employed people leaving work not too late) and night activities (for adults without children).

Once input all the required data, the Activity Generator creates, for each zone, trip chains made of home-based and non-home based trips. An example of trip chain is home to work, then work to shop, then shop to home. This is a far more realistic way to model trips during a day than the usual approach based on single trips. The disadvantage of this approach is the amount of required data: to achieve a good modeling of a large city thousand of zones are required, and for each zone all the previous data must be available.

Most of the outputs from SUMO are the same as the previous models. SUMO includes modules to calculate also noise pollution, but does not seem as good as other software with queues. Output statistics can also be defined for single vehicles.

Working on a larger scale than other models, SUMO does not models car parks.

Coming to kinematic models, SUMO is the software packages that has more options. Five models are available for use: the original Krauss model, the SUMOKrauss, an improved version of the Krauss model, the Kerner model, Treiber's Intelligent Driver Model and PW2009 (Paul Wagner's model using Teodosiev's action points). Each model uses different vehicle attributes: acceleration and deceleration capabilities for vehicle are used by all the models, then the Krauss models use a parameter called driver imperfection, while the other models focus on reaction time. This is modeled through different parameters: the IDM model uses two coefficients (k,φ), the headway and the minimum gap between vehicles (this last is a common parameter to all the models), while the other models use a reaction time coefficient called τ.

Finally, an original SUMO feature is the possibility to define the initial speed and the release lane for the vehicles. This is an option that S-Paramics and VISSIM do not have. On the other hand, the random generation of SUMO is probably weaker than the ones from other software, because it cannot keep in its memory the unreleased vehicles queue and, if a vehicle cannot be released, it is eliminated from the simulation.

4. CASE STUDY

4.1. The Subway Line M4 in Milan

The Subway line M4 in Milan represents the first East – West connection of the subway network in the city. The M4 will be an automated light metro, stretching for 15 km from Linate Airport in the East to S. Cristoforo railway station in the West. It will exchange with the other subway lines at S. Babila (M1), Croccetta-Policlinico (M3) and S. Ambrogio (M2). The connections with the surface suburban railway system will be at Forlanini FS, Dateo and S. Cristoforo stations. The estimated cost of the infrastructure will be 1.7 billion euros. The construction of the first part from Linate to Forlanini FS should start by mid-2012 and be completed by 2015 before the Universal Exposition. The other parts should be ready in the following years.

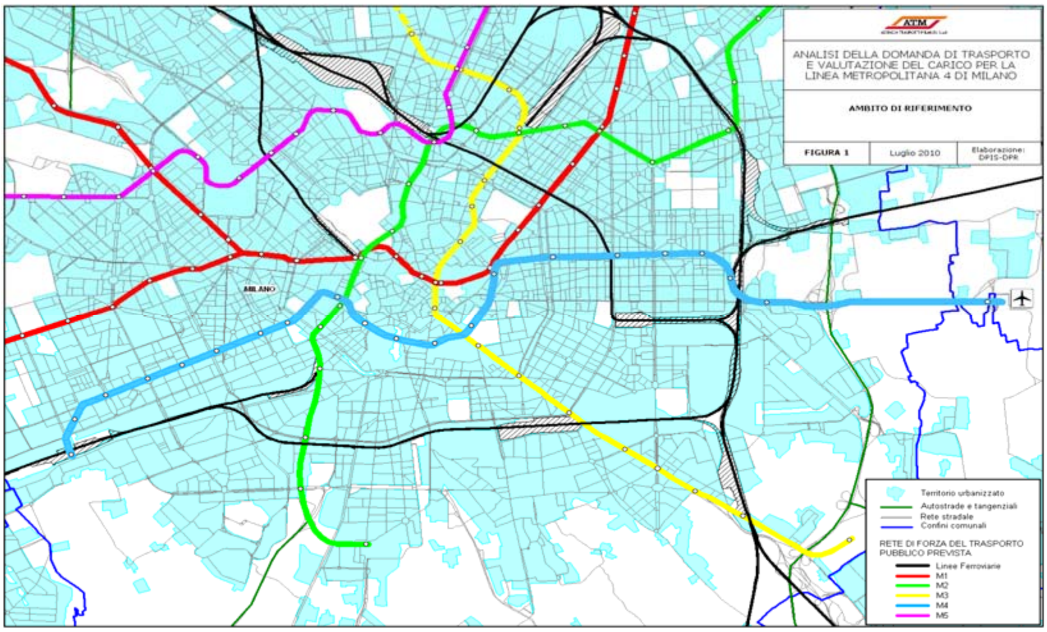


Figure 17: the ATM map of Subway M4 (light blue); M1 in red; M2 in green; M3 in yellow; M5 in magenta; in black, the suburban railway network

The line will develop along a very important urban corridor, the only one left without a major public transport infrastructure. In detail, the line will follow, from west to east, via Lorenteggio, the south west part of the medieval wall circuit called Cerchia dei Navigli, the corridor viale Indipendenza – Argonne and finally viale Forlanini to Linate Airport. This route represents also a connection to some important suburbs in the south-west of Milan, that will fall into the catchment area for the infrastructure thanks to park & ride facilities, such as Corsico, Buccinasco, Cesano Boscone and Trezzano sul Naviglio. These are also destinations of possible future extensions of the line.

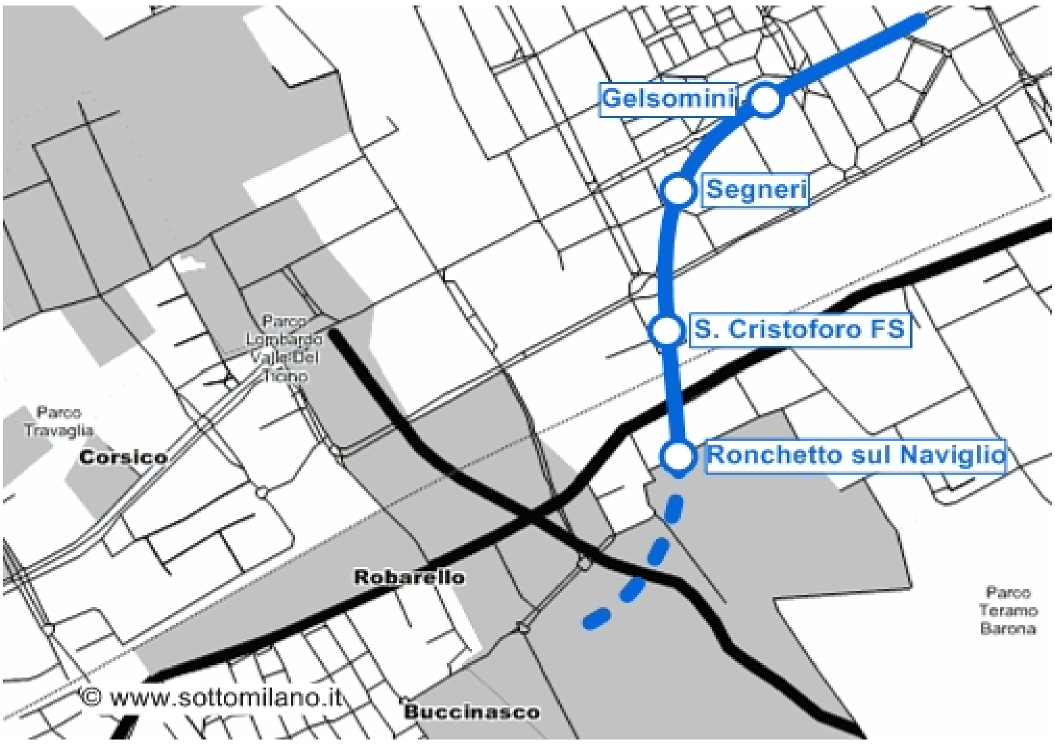


Figure 18: the M4 extension to the south west

On the east side, Linate Airport with its parking facilities can extend the catchment area for the M4 to San Felice, Segrate, Pioltello, Limito and Seggiano. This is the route of another possible extension.

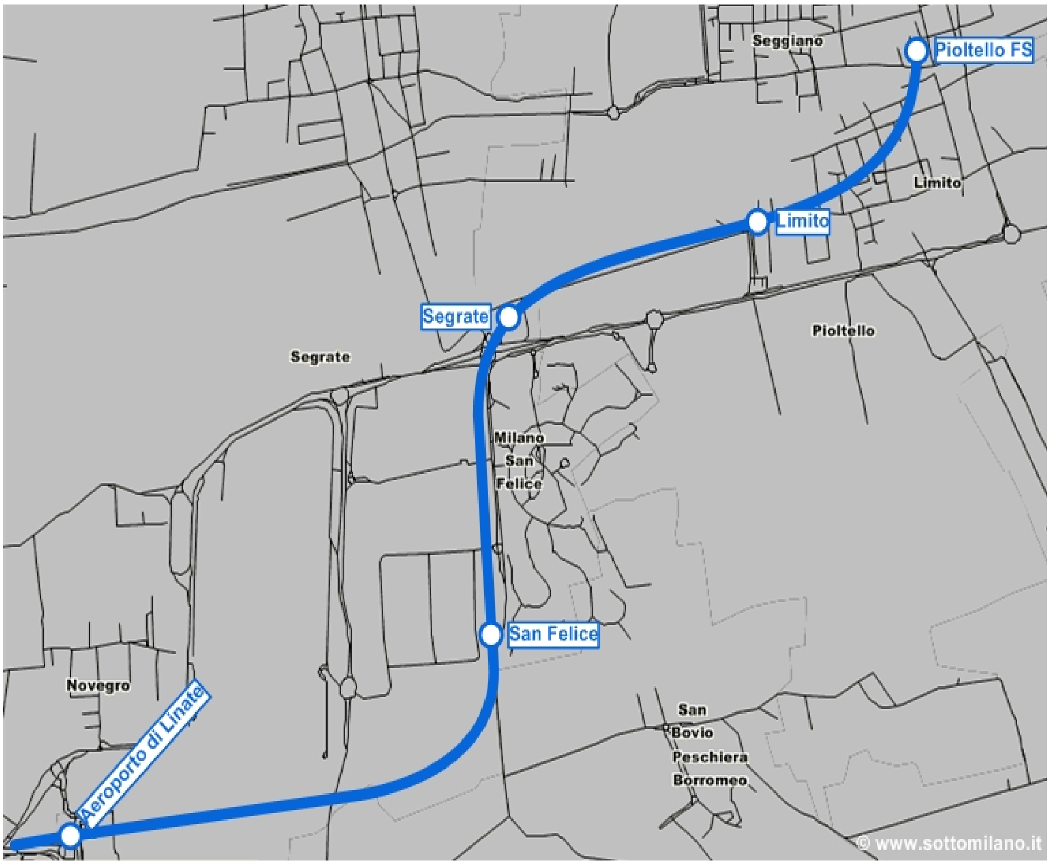


Figure 19: the M4 the line extension in the East past Linate Airport

Finally, a branch of the line may leave from Forlanini FS to reach San Donato (connection with M3) and San Giuliano in the far south of Milano.

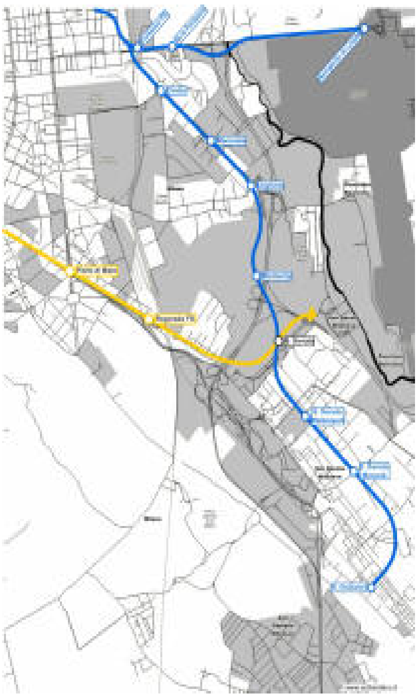


Figure 20: south extension of the M4 line

Anyway, considering only the urban part without the extensions, the only part of the line designed and funded so far, it passes through a dense urban environment bounded by important transport infrastructures, even if not served by any of them. In fact, north of Lorenteggio, along via Legioni Romane and up to piazza Pagano there is a branch of subway M1, while south of Lorenteggio there is the suburban railway S9 and the light rail 14. In the east of Milan, the M4 corridor is bounded by the light rails 12 and 27 in the south and light rail 5 and subway M2 in the north. None of these infrastructures, anyway, directly covers the M4 corridor.

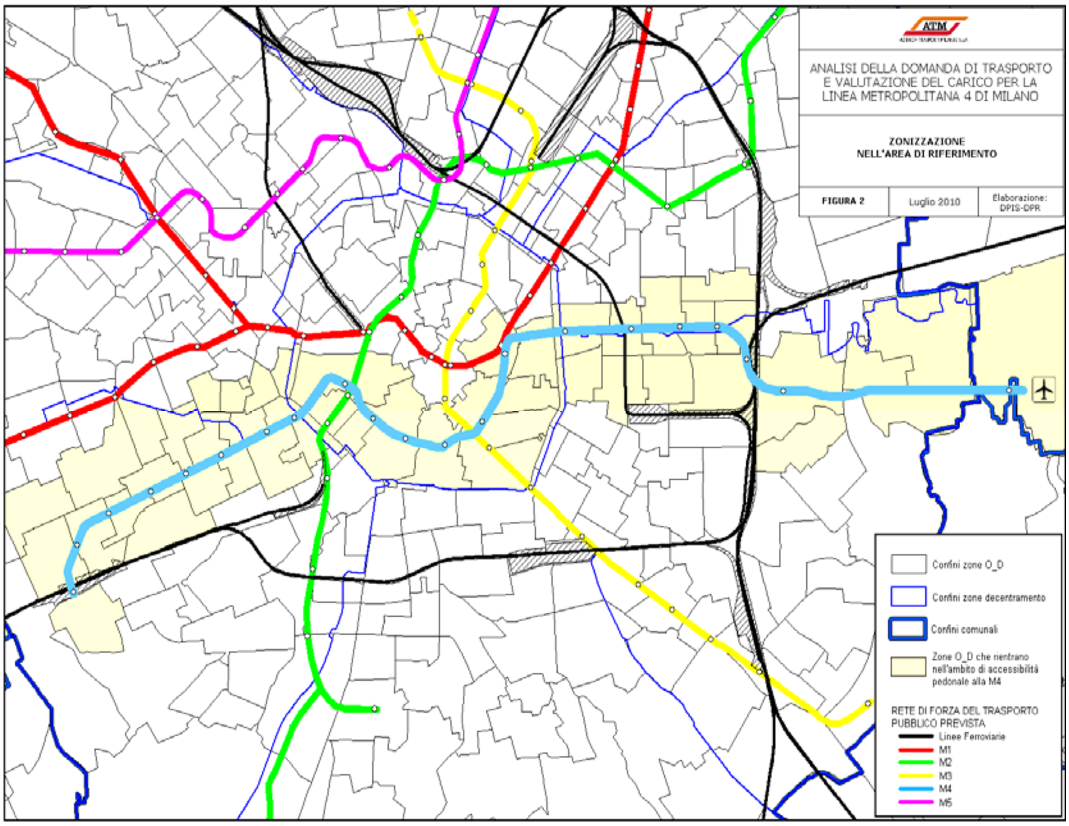


Figure 21: the M4 corridor with the zones included in the catchment area of stations (500m)

Considering for the stations a catchment area of 500m, the population served by the new line is around 200.000 people, 160.000 of which living in Milan and the others coming from the suburbs through different public transport or leaving their cars in the park & ride facilities. The population is equally distributed among the two parts of the line, east and west. It is expected a growth in the population of Milan in the next years, reversing the decreasing trend of the past decade. Currently, the highest density of population is located on Lorenteggio and Indipendenza – Argonne axes, exactly on the route of the M4.

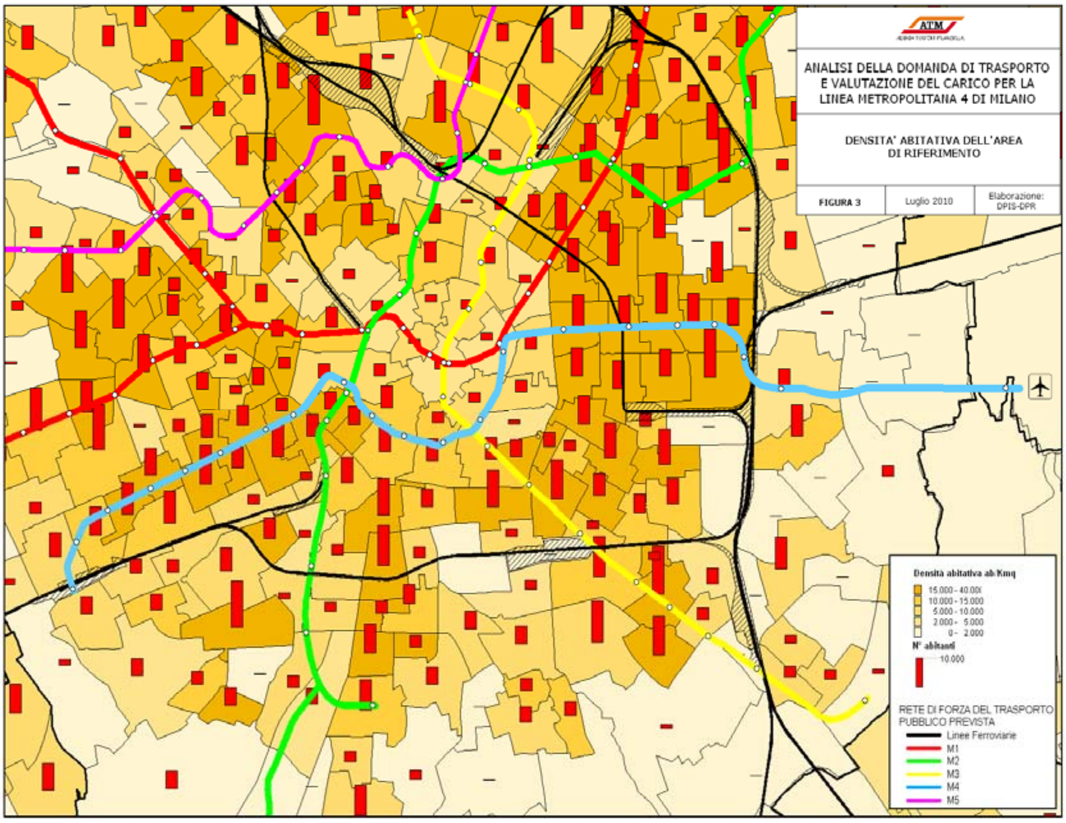


Figure 22: population density along the M4 corridor

The biggest part of the job opportunities along the new subway line is located in the city centre, between the S. Ambrogio and S. Babila stations. Most of these jobs are in the tertiary. Because the biggest residential areas are located, as discussed before, along the east and west part of the line, this confirms the radial structure of trips (from the suburbs to the city and vice versa) in Milan and along the M4 corridor.

Along the line are located as well very important attractors, such as two hospitals (Macedonio Melloni and Policlinico), two universities (State University and the Leonardo campus of Milan Polytechnic) and many high schools. The Linate Airport, at the east end of the line, represents the city airport of Milan and has 8.3 millions of passenger per year: it is evident the advantage of a direct link to the city centre for travelers that are mainly business visitors.

According to the planning documents of Milan City Government, many new developments are going to be realized along the line, in an attempt to densify and distribute traffic attractors and generators along the main transport infrastructure, in order to reduce the private traffic. On the west branch of the line, in the S. Cristoforo and Porta Genova stations redevelopments are planned 50.000 square meters of new constructions; 196 new apartments in social housing near Giambellino station; the Calchi-Taeggi development with 120.000 square meters of residential, commercial and tertiary buildings; Parri-Parco dei Fontanili development with 50.000 square meters of houses and the Savona-Brunelleschi development with 20.000 square meters of houses. On the west branch, the Porta Vittoria station redevelopment with 70.000 square meters of offices, hotels and the European Library for Information and Culture and the via Cena development, with 3000 square meters of residences. The total sums up to 310.000 square meters of new developments.

Figure 23: map of M4 with the new developments (orange), universities (green) and hospitals (red) along the corridor

The current situation, excluding the new developments, creates a demand, along the corridor, of 650.000 trips, 400.000 of which attracted and 250.000 generated. This data comprises all the relations developing to and from the considered area, even if they not follow the subway corridor. The modal split is currently in favor of the car.

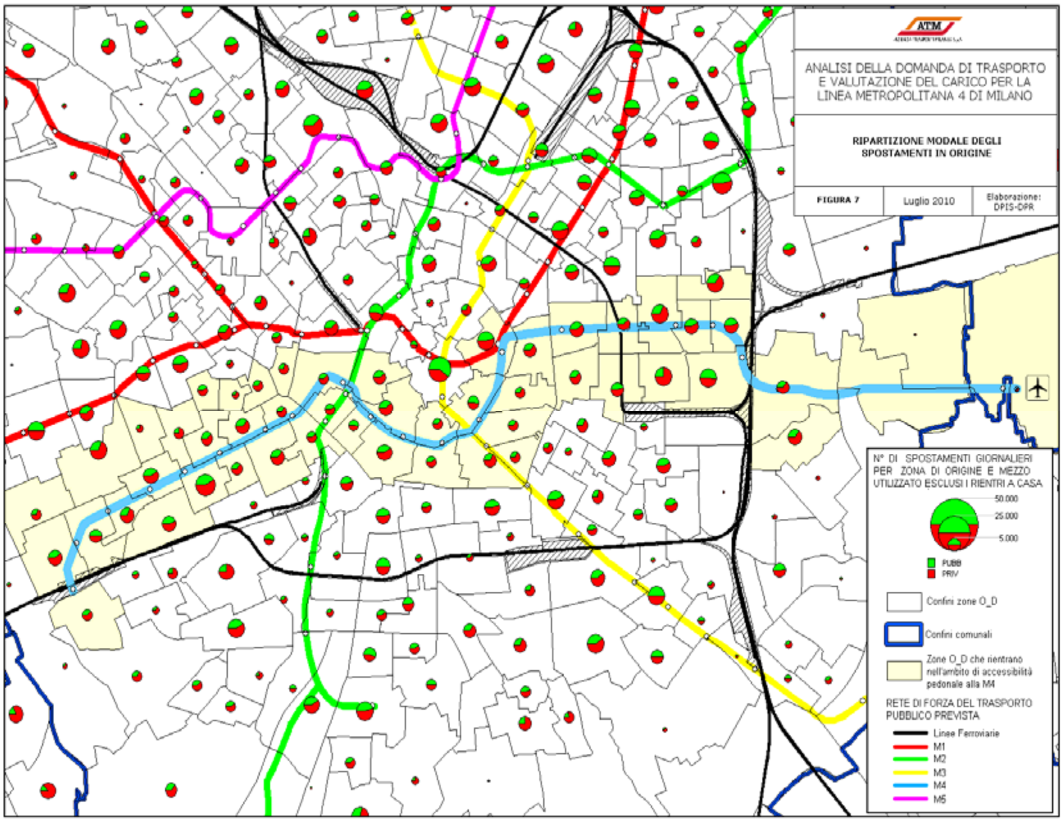


Figure 24: current (without M4) modal split of generated trips along the corridor; in red private transport share, in green public transport

From a confrontation with neighboring corridors already served by heavy transport infrastructure, it is evident the difference in the modal split, especially keeping in mind that this is the central area of Milan, where public transport has an high level of service, traffic is in general congested and parking is difficult to find or very expensive. Wherever a rail, subway or light rail line is present, the modal share for public transport is much higher than in the M4 corridor, showing an high potential appeal of the line the users. In fact, studies carried out for other public transport projects in Milan estimated modal shifts of 5-8% towards the improved public transport systems, with peaks of 10-20% for trips with origin or destination along the new railway or subway lines. The demand estimation for the M4 has been done anyway with smaller values to avoid the overestimation of the demand.

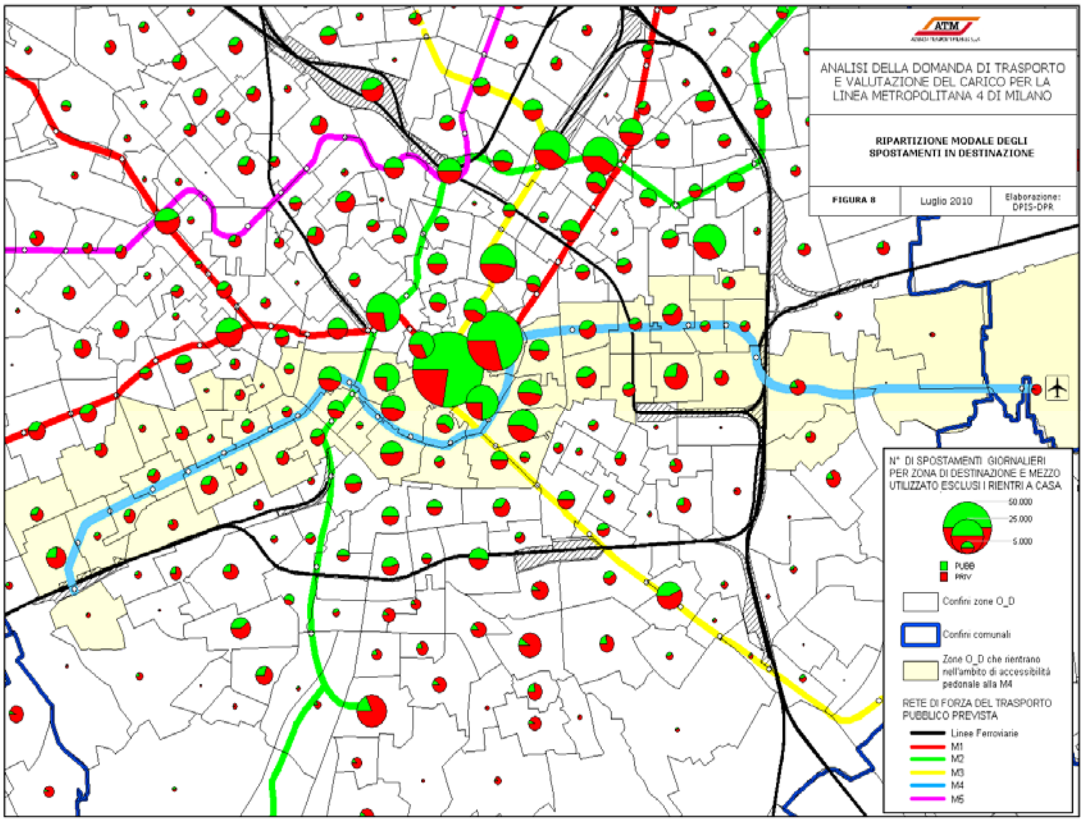


Figure 25: current (without M4) modal split of attracted trips along the corridor; in red private transport share, in green public transport

The current public transport services on the M4 corridor are run with buses. The commercial speed is 13 km/h, the offered capacity is 39.000 persons per day per direction. From surveys, passengers are 25-30.000 per day per direction, with an occupancy between 60 and 80% in peak hours. The design commercial speed of M4, 30 km/h, will for sure attract many more passenger on public transport along the corridor among the people that today decide to travel by car.

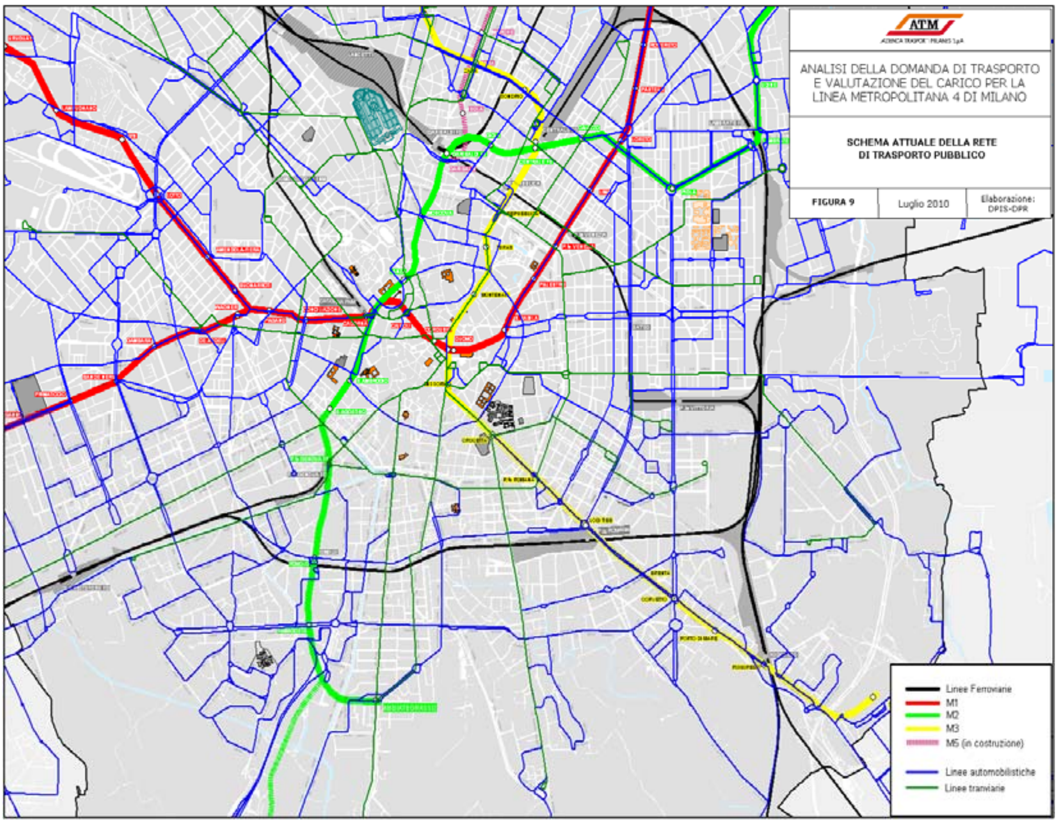


Figure 26: current public transport network; in blue the main bus routes, in dark green the light rail lines

The presence of the new infrastructure would anyway require a re-organization of the public transport along the corridor, with the elimination of the radial lines along the M4 route and the improvement of transversal lines with a feeder function for the new fundamental infrastructure.

The result of all these studies are the loading diagrams presented in the next page. The most loaded branch is the west one, with a maximum load in S. Ambrogio of 9.000 passengers/hour in the current scenario and 13.000 passengers/hour in the future scenario. The load then decreases after the crossing of the city center. On the opposite direction, the pattern is similar, with 6.000 passengers/hour in the current scenario and 9.200 passenger/hour in the future. The number of passengers on this branch is probably underestimated due to the presence of the airport, whose non-systematic traffic is very difficult to estimate, especially in case of introduction of such a big change in transport supply with the substitution of bus lines with a subway line.

The total number of passengers on the line in the AM peak hour is estimated to be 29.000 passengers in the current scenario and 44.000 in the future. According to the data available for the other subway lines, this means 200.000 passengers/day and 60 million passengers/year in the current scenario, becoming respectively 300.000 passengers/day and 90 million passengers/year in the future scenario.

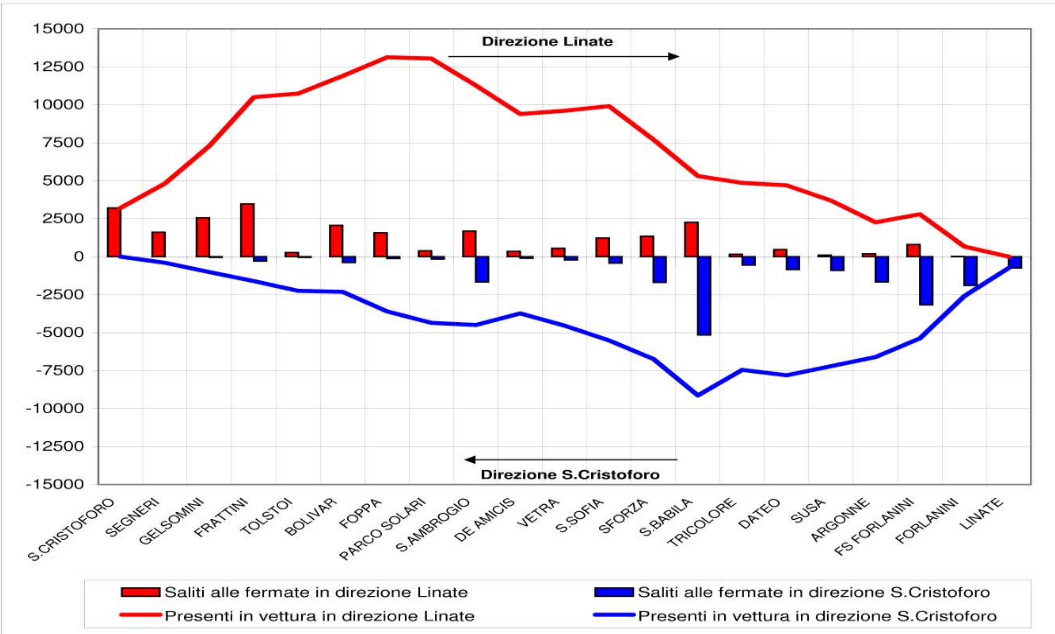


Figure 27: current demand AM peak hour loading diagram for the M4; in red eastbound direction, in blue westbound direction

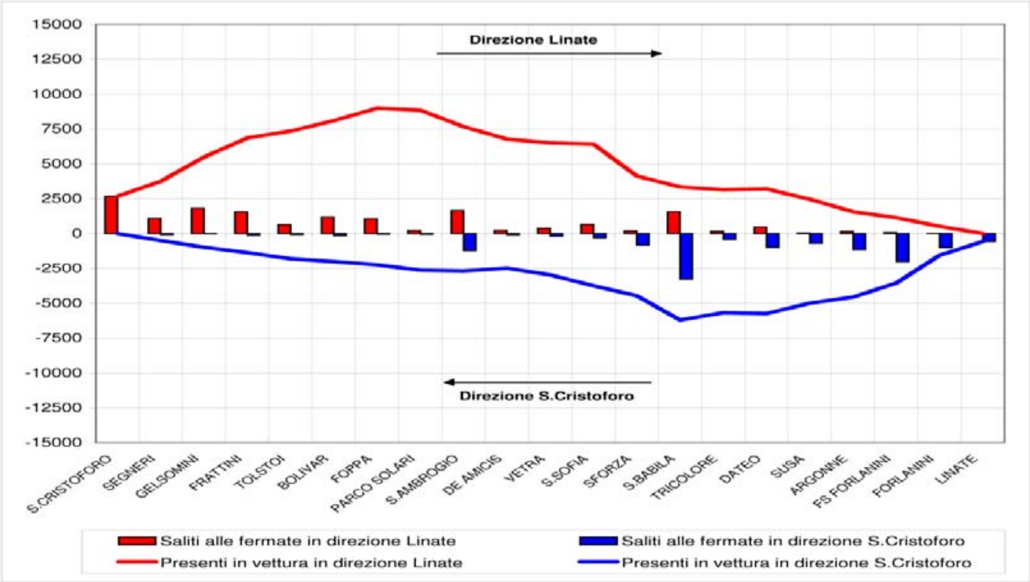


Figure 28: future demand AM peak hour loading diagram for the M4; in red eastbound direction, in blue westbound direction

4.2. Data Collection

Within this major infrastructure project, this case study involves the verification, through a micro-simulation model, of the impact of construction operations and building site of the Quartiere Forlanini station. The station will be directly below viale Forlanini, a major road axis in Milan, connecting Linate Airport, the East Ring Road of Milan and the city centre. The section of viale Forlanini has 3 lanes per direction and a bus lane in the middle of the road. The station will be built digging a crater from the road level down to the tunnel level. With a flow of around 2000-2500 vehicles per hour on each direction in the peak hour, the building site will have a massive impact on traffic.

In order to start the modeling activities, traffic data were needed. On February 13th 2012 we carried out a traffic survey from two observation points using cameras, from 7.30 to 8.30 AM. We surveyed two related intersections: between via Forlanini and via Cavriana and between viale Forlanini and via Repetti.

Figure 29: building site situation with observation points in red and the area of the station in orange

The survey was necessary because the data obtained from the macro-model of Milan were not reliable according to the calibration we did considering the fixed survey stations of the Milan traffic office around the city. We identified the cause of these problems in the introduction of a congestion charge in the city center, that significantly changed the flows in the city and even the mobility patterns in urban area. Because of time constraints, we could not re-calibrate the model, so we chose to run on site surveys.

The results of the survey are shown in the image below. As anticipated, viale Forlanini has a flow of around 2500 vehicles per hour going to Linate Airport, and around 1500-2000 directed to the city centre. One of the big problems of this intersection is the presence of important left turn maneuvers, in the order of hundreds of vehicles, that require dedicated phases, extending the cycle length. The left turn from viale Forlanini to via Repetti has also many heavy vehicles, coming from the Milan Ring Road and going to the fruit and vegetable Milan central market.

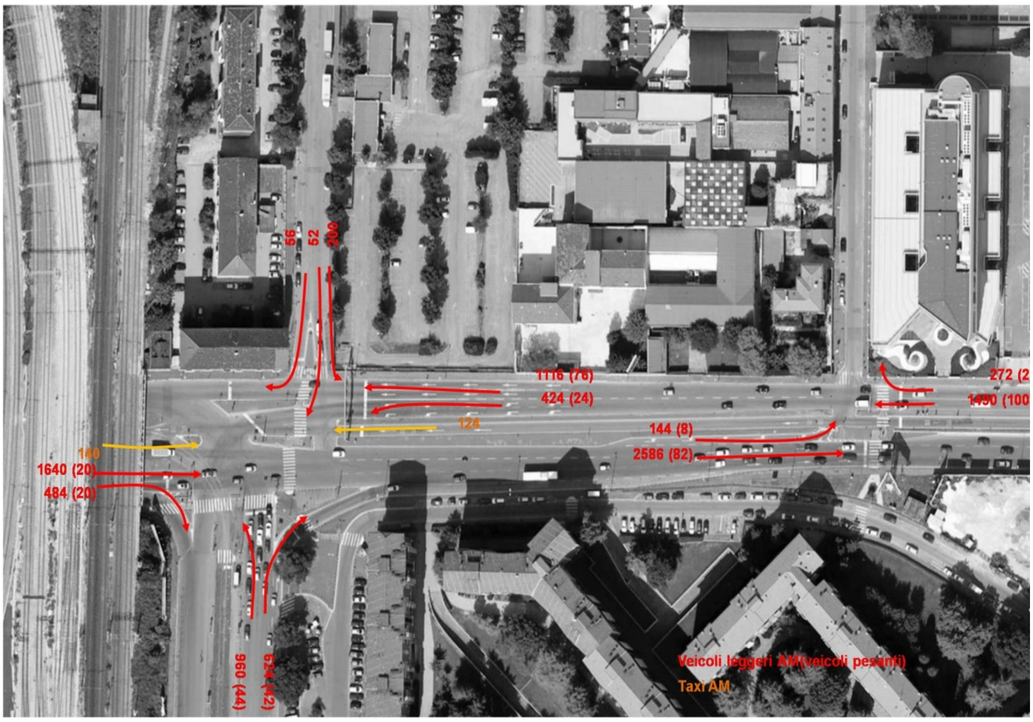


Figure 30: observed flows in the area of the station (heavy vehicles between brackets)

Once the data from the survey records were registered, it was easy to create and O/D matrix. Actually, in this case we could have assigned the vehicles through the itineraries approach (see chapter 2.1.1), because in this simple network there is no route choice. Anyway, we created 4 zones at the boundaries of the network: in the south, along via Repetti; in the north, at the intersection between via Gatto and via Cavriana, the first being a one way street entering the intersection and the second leaving the intersection to the north; in the west a zone along viale Forlanini on the city side and in the east a zone along viale

Forlanini on the Linate Airport side.

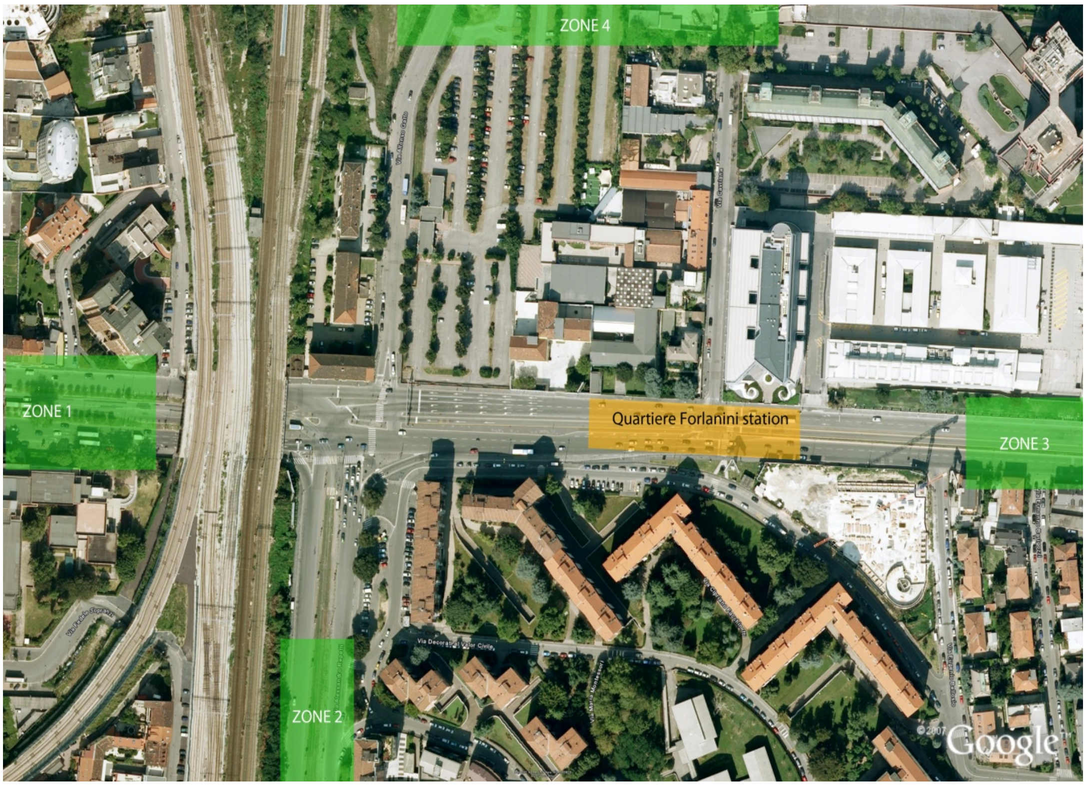


Figure 28: the zone system for the viale Forlanini subway station model

The O/D matrices result therefore

LIGHT VEHICLES

	1	2	3	4
1	0	484	1736	44
2	744	0	524	100
3	1116	424	0	272
4	56	52	200	0

HEAVY VEHICLES

	1	2	3	4
1	0	20	20	0
2	12	0	42	8
3	76	24	0	2
4	0	0	0	0

TAXIS

	1	2	3	4
1	0	0	140	0
2	0	0	0	0
3	124	0	0	0
4	0	0	0	0

Figure 29: O/D matrices for the Base scenario

4.3. Quartiere Forlanini Station Building Site Model

4.3.1. Modeling the Demand

With the O/D matrices available, it is possible to model the demand. This process involves the definition of three levels of matrix, the simulated time periods and the profiles.

The three matrices are the ones outlined in figure 26 and derive from the traffic survey. The Light Vehicles Matrix comprises cars and commercial vehicles, with a proportion of respectively 95% and 5%. The Heavy Vehicles Matrix considers trucks and goods vehicles, with a proportion of 90% for Medium Goods Vehicles (two axles) and 10% Heavy Goods Vehicles (semi-trailer trucks). Finally, we created the Taxis matrix in order to simulate the use of the reserved bus lane by a relevant number of taxis (it obvious, the intersection considered being on the only route connecting the city centre to Linate Airport). We applied restrictions on relevant links in order to force the program to assign the taxis flows on the bus lanes, where available.

The public transport does not require a matrix to be simulated. Public transport lines are simulated creating the bus routes as fixed routes and defining a timetable for each through headway and time of the first ride.

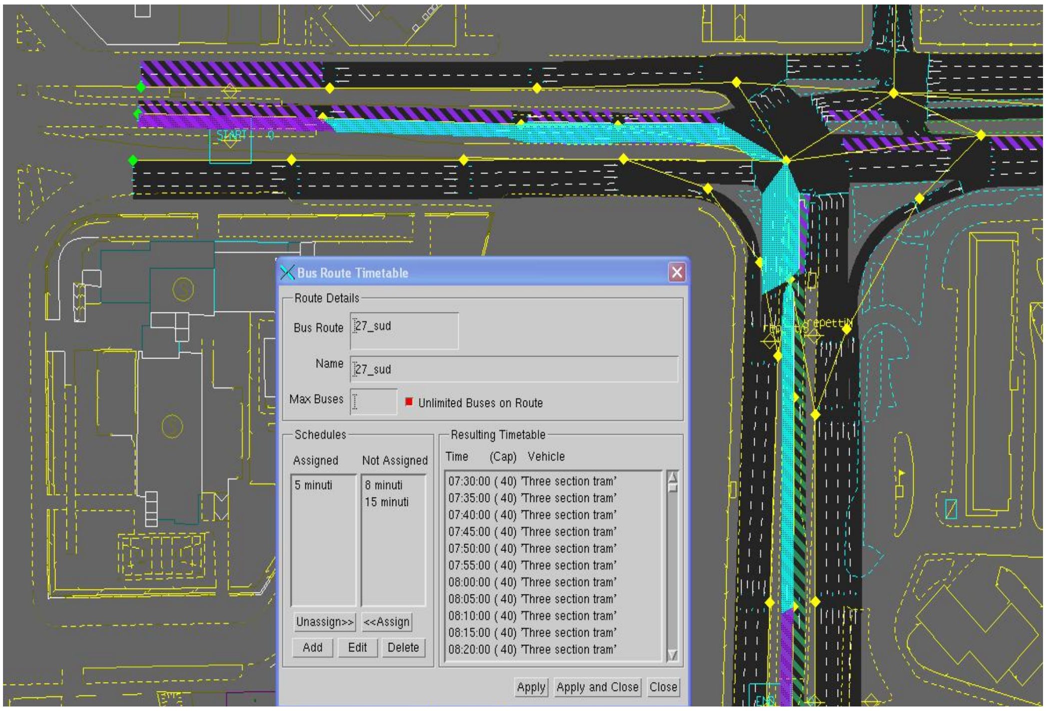


Figure 30: a screenshot from S-Paramics with a bus route highlighted and the Bus Route editor

In this case, the bus lines 45 and 73 and the light rail line 27 cross the study area. The bus 45 has an headway of 15 minutes, enters the area from zone 1 and leaves from zone 3 when moving to the suburbs, and moves from zone 3 to zone 1 when heading to the city centre. Bus 73 moves from zone 1 (city centre) to zone 3 (airport) and vice versa with an headway of 8 minutes. Light rail 27 goes from zone 2 to zone 1 and vice versa with a frequency of 20 rides per hour during peaks. (see figure 28 next page)

We simulated the same time period of the observations, from 7.30 AM to 8.30 AM. The time periods are 0.00-7.30 AM, 7.30-8.30 AM and 8.30 AM to 0.00 AM. The simulation starts at 7.29 AM, at the end of period 1, to allow for a small pre-loading period, and finishes at 9.00 AM. For each period all the three matrices are present, but only in the period 2 they are not empty matrices. The simulation goes into period 3 to let all the vehicles reach their destinations.

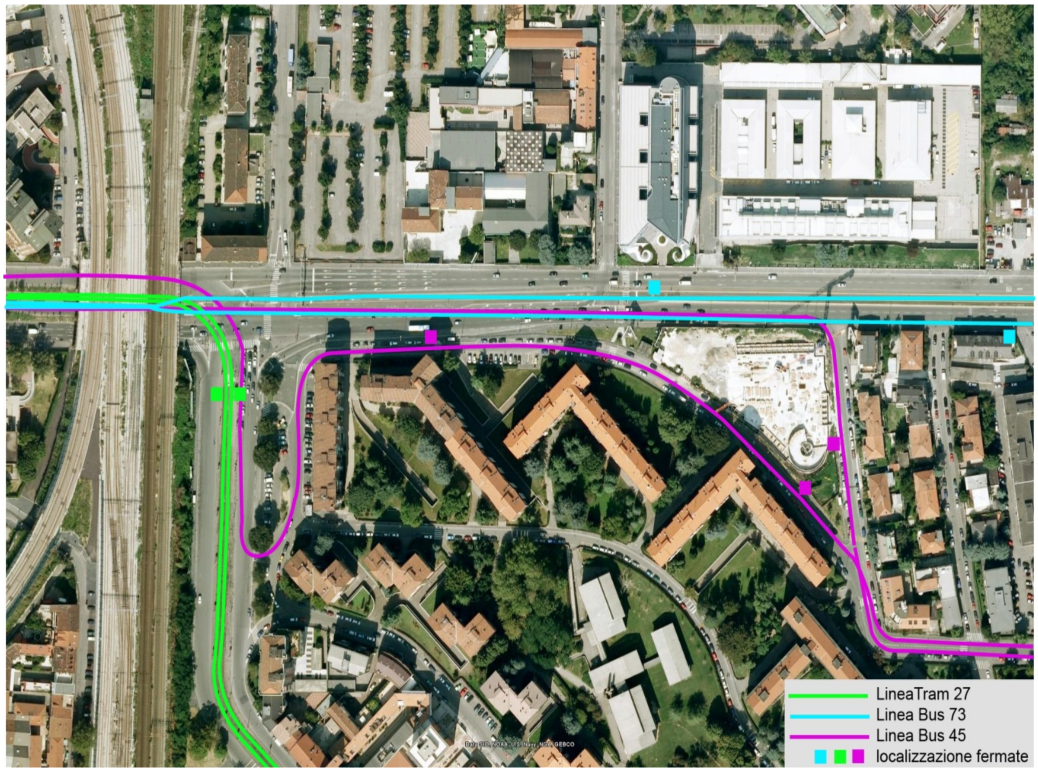


Figure 31: public transport lines and stops in the study area

We did not have data on profiles. We observed an increase of traffic close to 8.00 AM and a decrease close to the end of the observation period at 8.30 AM. To simulate these most critical 15 minutes, we manually created a profile increasing the percentage of released vehicles from 8.00 to 8.15 AM for all the O/D pairs. The only constraint in the editing of profiles is that the sum of the 5 minutes percentages of released vehicles during the simulation period must be equal to 100%.

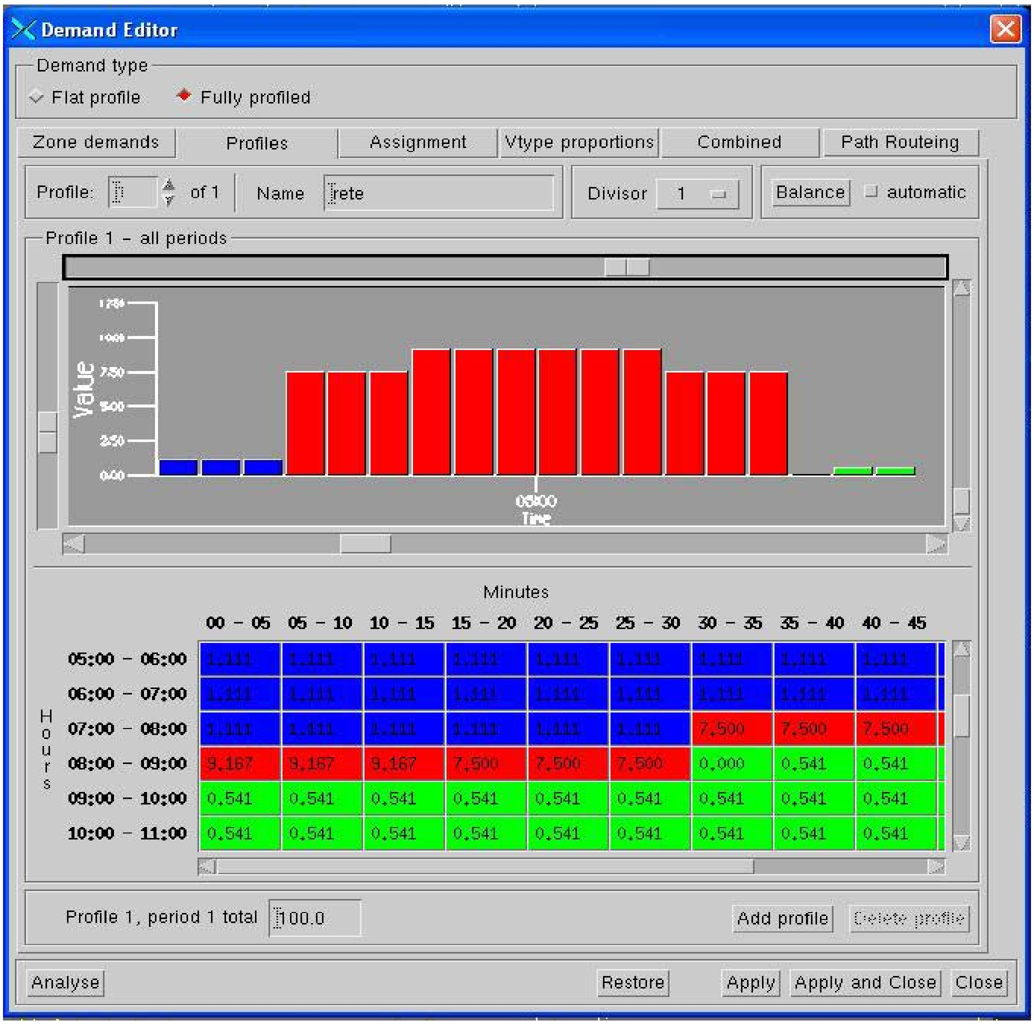


Figure 32: S-Paramics demand editor with the peak hour profile

4.3.2. M

4.3.2 Modeling the Infrastructure Supply

The road network in the study area is composed by multi-lane main roads, with carriageways separated by kerbs.

Viale Forlanini (east-west axis, from zone 1 to zone 3) has 3 lanes per direction with a central bus and taxi reserved lane westbound, towards the city centre. Viale Repetti, approaching the intersection from the south (zone 2) has as well 3 lanes per direction, with light rail tracks in the middle, separated from the other traffic by kerbs. The northbound roads, via Gatto and via Cavriana, are regular one lane, one way roads, via Gatto entering the intersection and via Cavriana leaving the study area.

The very first step is to load into the software a technical map of the study area in DXF format. This is used to create a realistic network both in spatial and street layout. To ease the construction, the map must be cleaned from all the unnecessary elements. The coordinates of the map must be changed and set as close as possible to zero in order to achieve a better functioning of the software.

Figure 36: the base technical map of the viale Forlanini - via Repetti intersection

We used separated sets of nodes and one-way links to model each carriageway of the main roads. A different set of nodes and two-way links was used to model the public transport reserved lanes in the middle of the streets. To simulate reserved lanes there are two ways: the “Bus Lane” botton automatically excludes all the vehicle types that do not belong to the public transport routes, i.e. allows only the vehicles assigned to a public transport route, defined as described in the previous paragraph, to use the lane. The other possibility is to manually set the allowed or barred vehicles types. We used both these approaches: for the lanes with only light rail tracks we used the “Bus Lane” button. The other lanes, shared as well by taxis, were defined using the manual setting of restrictions. We assigned the eastbound direction of route 45, from zone 2 to zone 1, to the shared carriageway so that the model would not generate a bus on non-sealed light rail tracks. The last network element for public transport are bus stops, that can be attached to a link at any position. The default of the software is the median position, but the user can move them from node to node. Each bus stop gives the user the opportunity to define the stop for all the lines passing on the link, or only to some of them.

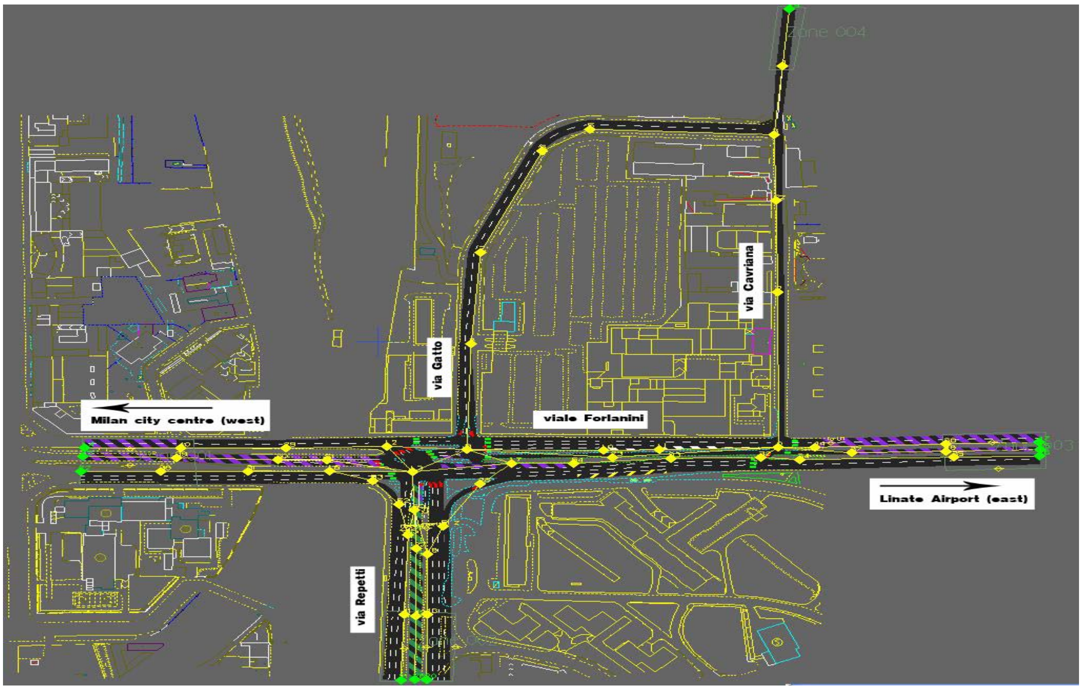


Figure 33: the complete network with links, nodes, bus lanes (green filling), lanes with restrictions (purple filling), zones and connector nodes (green dots) and traffi lights

This representation of separated carriageways made the simulation more realistic but the modeling of intersection more complicated. The main intersection between viale Forlanini and via Repetti is itself complicated in reality, with many different interacting maneuvers and a large number of kerbs and traffic islands to separate them.

The small intersection between viale Forlanini and via Cavriana was the easiest to model. In the figure, Node 32 represents the separation of left turn reserved lane from the eastbound direction of viale Forlanini, crossing the kerb of the bus lane. Node 0 represents the intersection of this lane with the bus lane and the westbound direction of viale Forlanini. The traffic lights are placed at the of link 33-32 for the eastbound carriageway and on the link 44-0 for the westbound traffic light. To model it we had to unify in the same intersection the nodes 32 and 0, so that the link connecting them was considered as internal to the intersection and no traffic lights were created on this link. If we considered the two nodes as separated, a traffic light would have been created on the 32-0 link, and even if it was possible to set the same cycle, the unrealistic situation of a vehicle stopped on this link, occupying the bus lane, could have happened if the vehicle stopped at the amber of the second traffic light after passing the first one at the last moment.

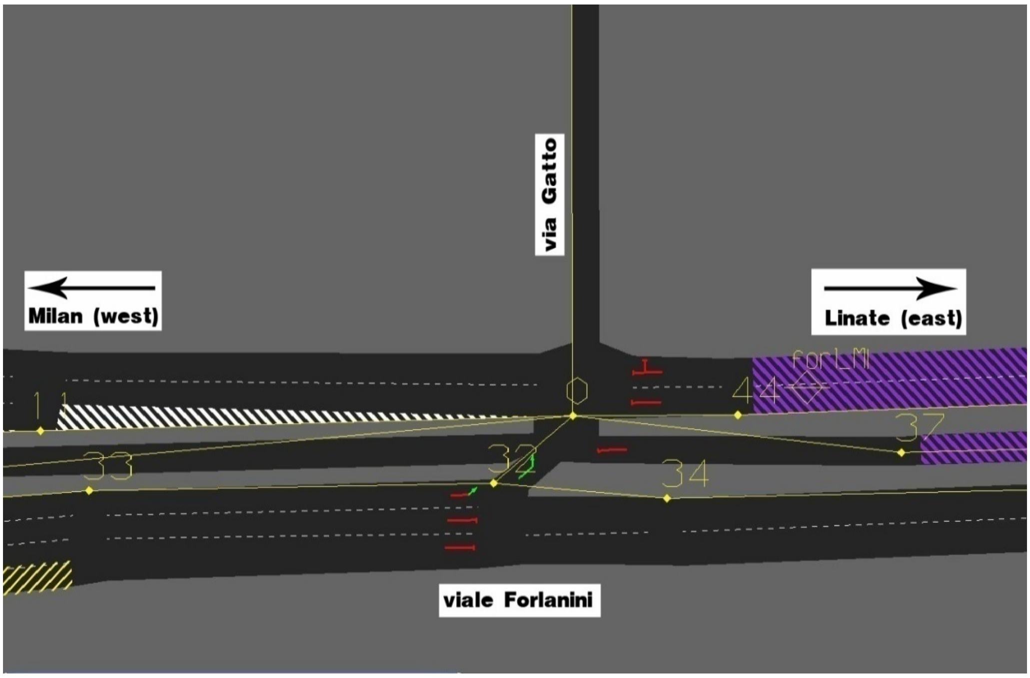


Figure 34: the via Cavriana intersection model

The viale Forlanini – via Repetti intersection was more problematic. It was impossible to simulate the intersection as a single node. We split the intersection in three different junctions. Nodes 16 and 17 represented, respectively, the separation of the right turn lane to via Repetti from Milan city centre and the merging of the right turn lane from via Repetti to Linate Airport.

Node 9 represents the intersection between the westbound direction of viale Forlanini, considering both the private traffic and public transport carriageways, and via Gatto. Node 9 is the origin of three links: one (9-12, two lanes one way) to simulate the two lanes of the straight movement of viale Forlanini towards Milan city centre, used also by the vehicles turning right from via Gatto to the city centre; the second (9-13, two lanes one way) is shared between the vehicles turning left from viale Forlanini to via Repetti and from via Gatto to via Repetti; these two movements are separated by a traffic island. The connection 9-17 (one lane one way) allows the vehicles from via Gatto to complete the left turn on viale Forlanini towards Linate Airport.

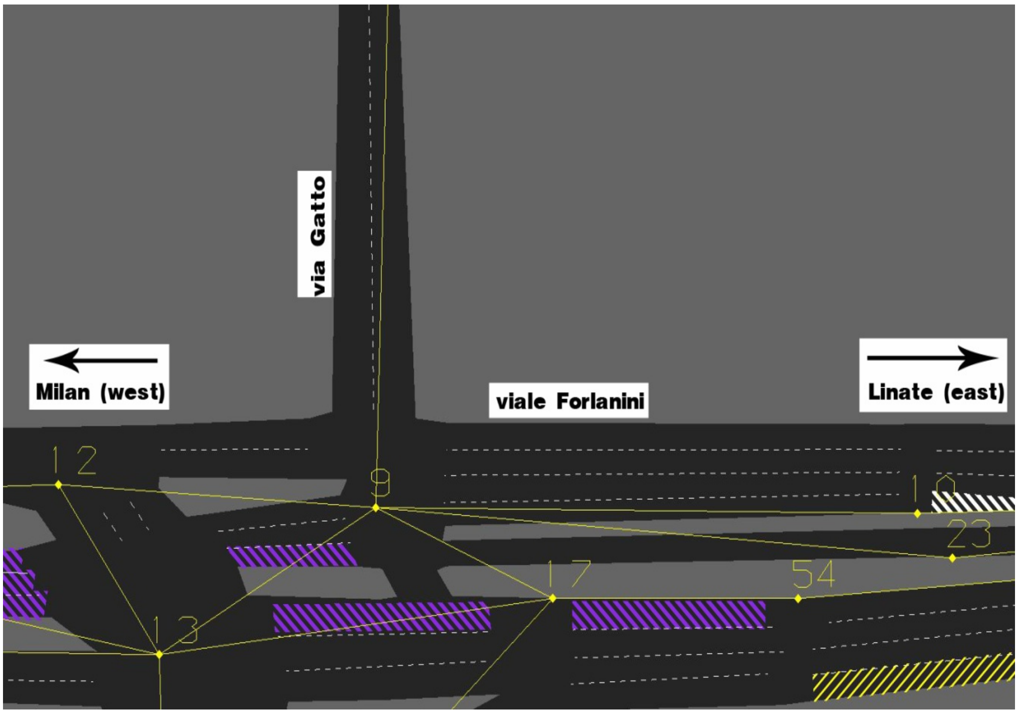


Figure 35: the node 9 model

This configuration causes the software to create a left turn from the westbound direction of viale Forlanini to the eastbound carriageway, obviously unrealistic. This maneuver needs to be barred in the "Junction" step of intersection modeling. Also, the movement from node 23 (bus lane) to node 12 through node 9 must be barred because it is not allowed in reality even if the intersection model configuration makes it possible. At the same step, maneuvers are assigned to available lanes as shown in the image below.

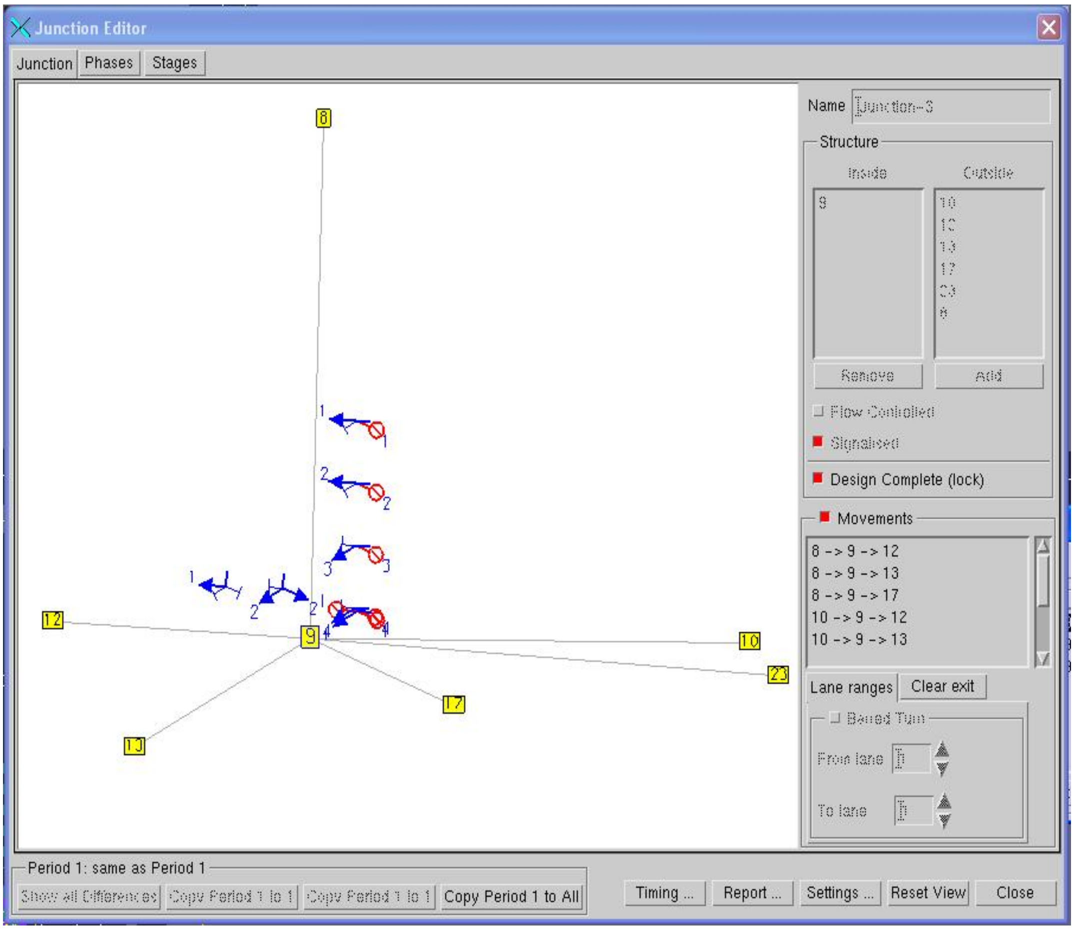


Figure 36: junction editor in S-Paramics for node 9

In the “Phases” step, a letter is assigned to each group of maneuvers that are allowed to move at the same time. Because this intersection has a lot of offsets, in this case pretty much each maneuver is related to a letter, but for simpler intersections this is not always the case.

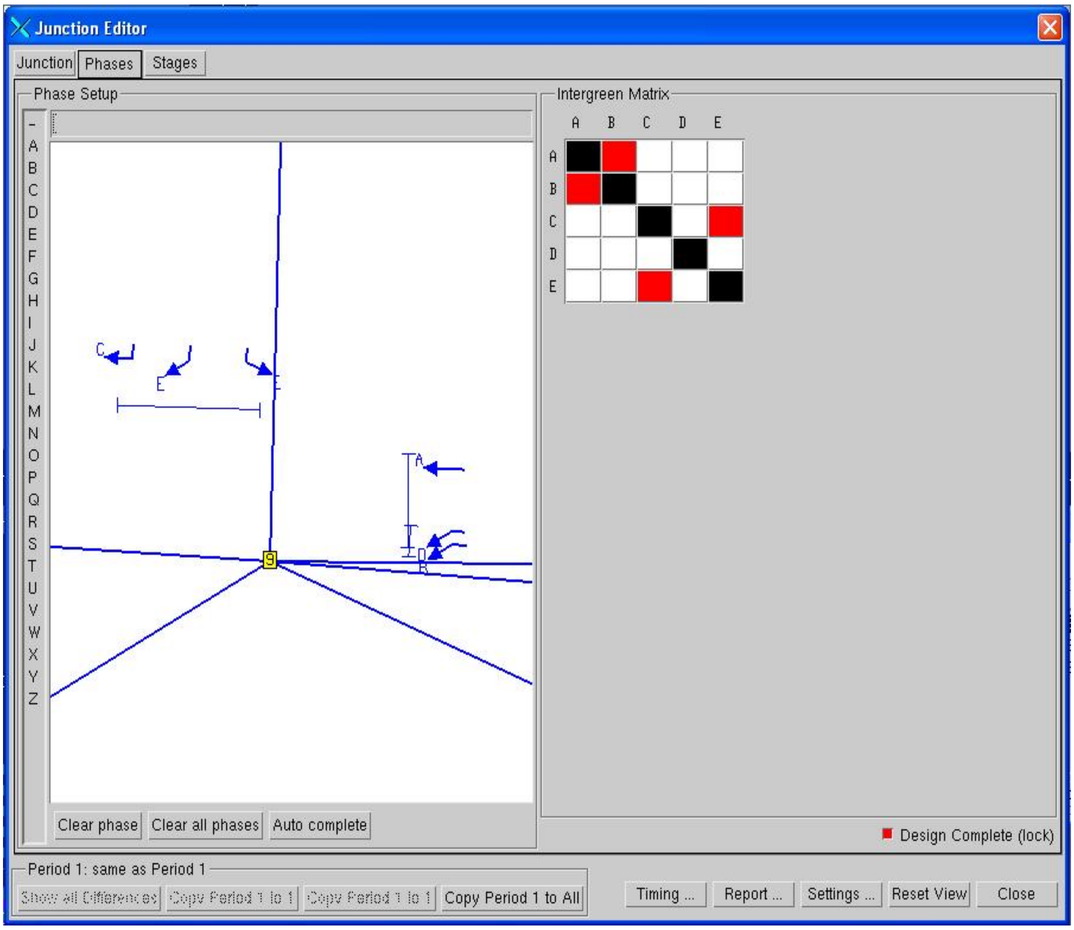


Figure 37: the Phases editor of node 9

In the final “Stages” step, to each phase a green, yellow and all red time are assigned. Phases A (straight from viale Forlanini) and B (straight from the bus lane) have 66 seconds of green in stage 1. Phase B has an offset of 10 seconds with respect to phase A, that is, B starts 10 seconds later than A. In Stage 2, phases C and E (maneuvers from via Gatto) have 48 seconds of green. Stage 3 has 19 seconds of green for stage D, the left turn from viale Forlanini to via Repetti.

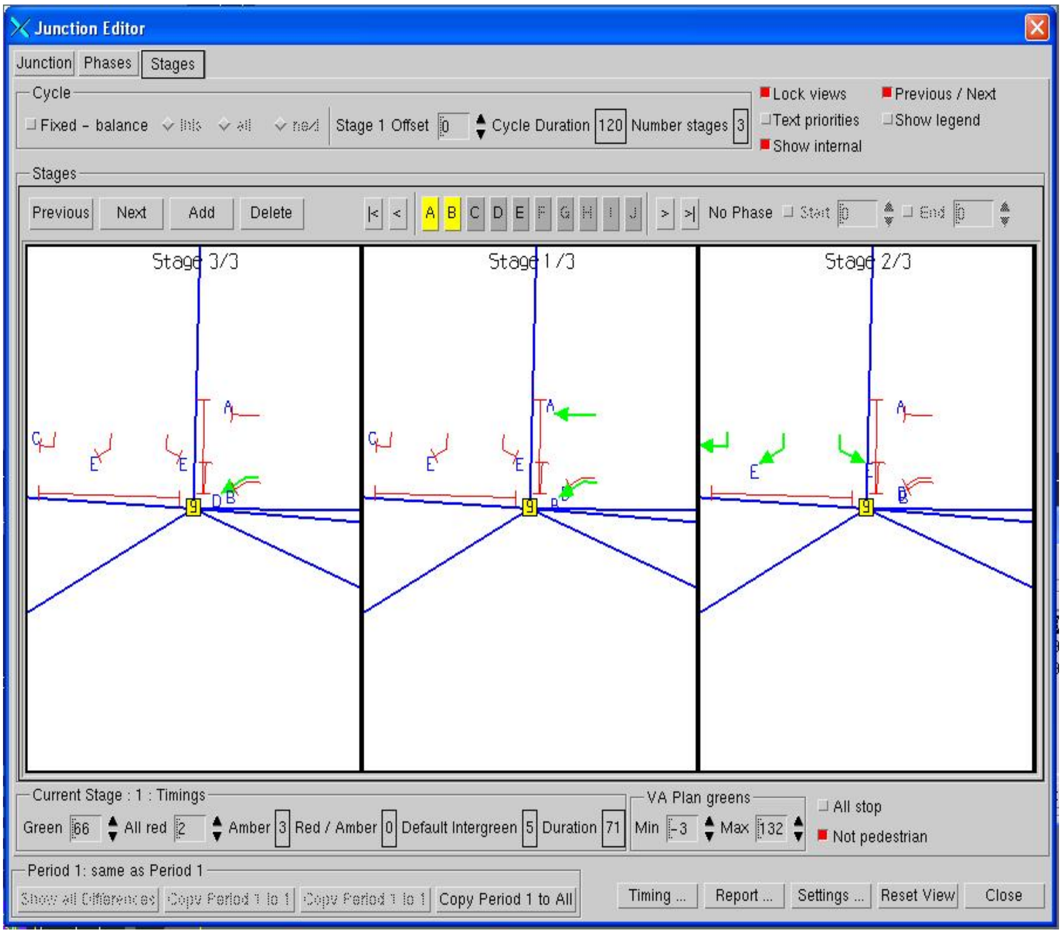


Figure 37: stage editor for node 9

Node 12 *represents* the merging of the maneuvers moving towards the city centre coming from node 9 (straight from viale Forlanini and right turn from via Gatto) and the left turn maneuver from via Repetti to via Forlanini westbound. The only link originating from this intersection is the westbound carriageway of viale Forlanini.

All the possible maneuvers at this intersection are actually allowed. There are only two stages, one for each maneuver. In the real traffic light there are only two stages, one for each maneuver, and a long all red period. In the model, to coordinate this cycle with the others, we created three phases, the first one equal to Stage 1 of node 9 with green for straight maneuver, the second equal to stage 2 of Node 9 with green for the left turn maneuvers, and a third one with enough green to complete the cycle; this third green, that does not exist in reality, is useful also to avoid having vehicles unrealistically stuck in the middle of the intersection, especially the ones leaving via Repetti at the end of the green time that may not finish the intersection crossing before the end of the stage.

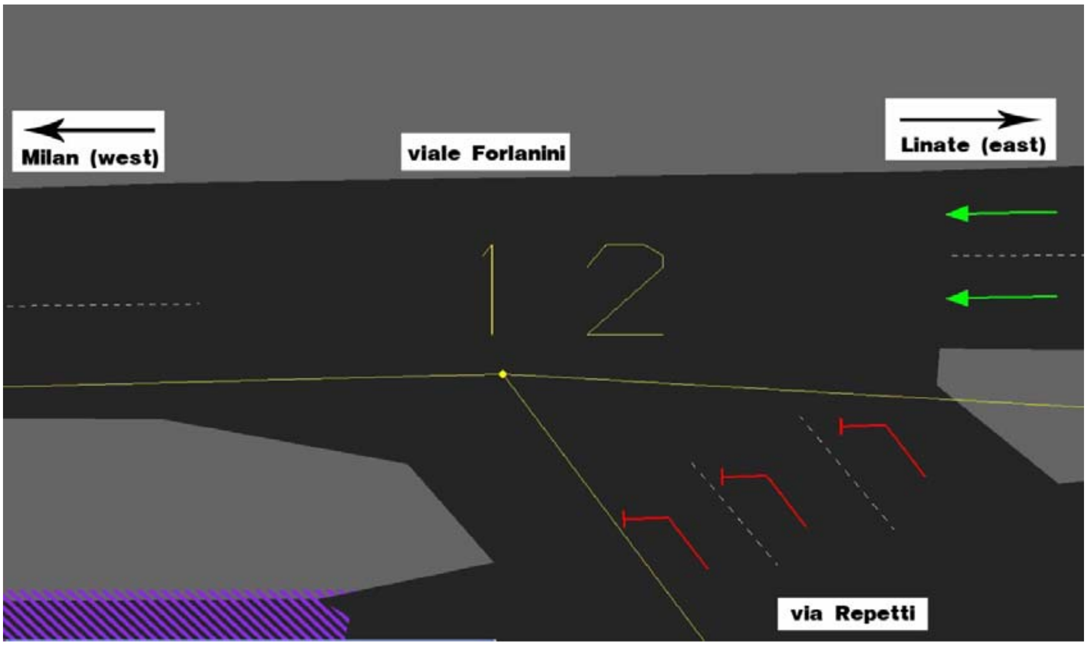


Figure 38: node 12 stage 1

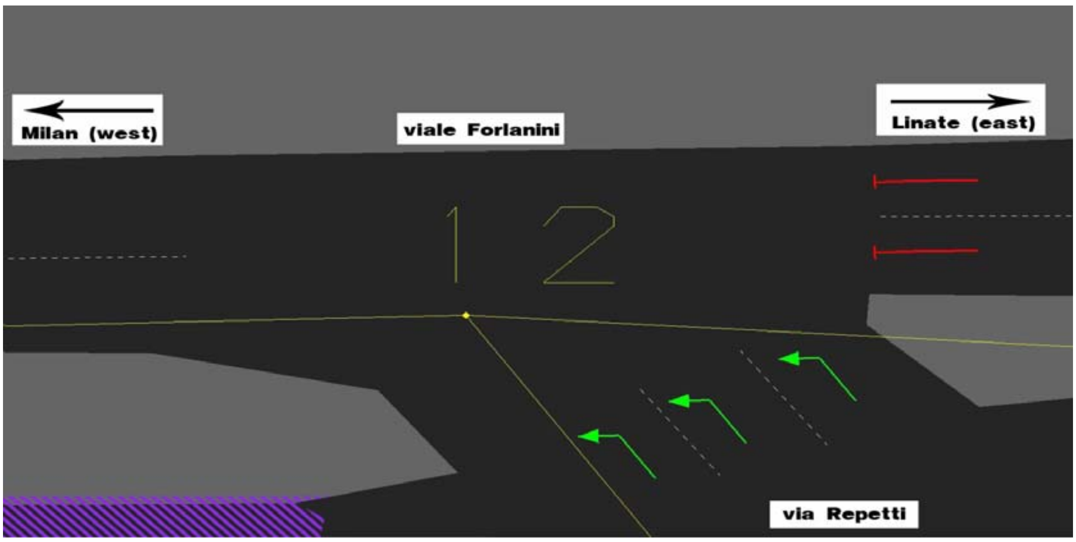


Figure 40: node 12 stages 2 and 3

The final node of the intersection is node 13. It represents the junction of the bus lanes going eastbound to Linate Airport from the city centre, the eastbound carriageway of viale Forlanini, the lanes carrying the vehicles turning left from viale Forlanini westbound and via Gatto to viale Repetti (link 9-13) and viale Repetti itself. The node allows vehicles from viale Forlanini (both private and public transport) to proceed straight to Linate, the vehicles from Linate Airport and via Gatto to turn into via Repetti and the vehicles from viale Repetti to turn left to viale Forlanini to the city centre (link 13-12, three lanes one way) to viale Forlanini. Because of the presence of light rail tracks, the approach lanes of the public transport reserved lanes are switched, with the external one used for crossing maneuvers and the internal one for light rail right turn (link 51-13).

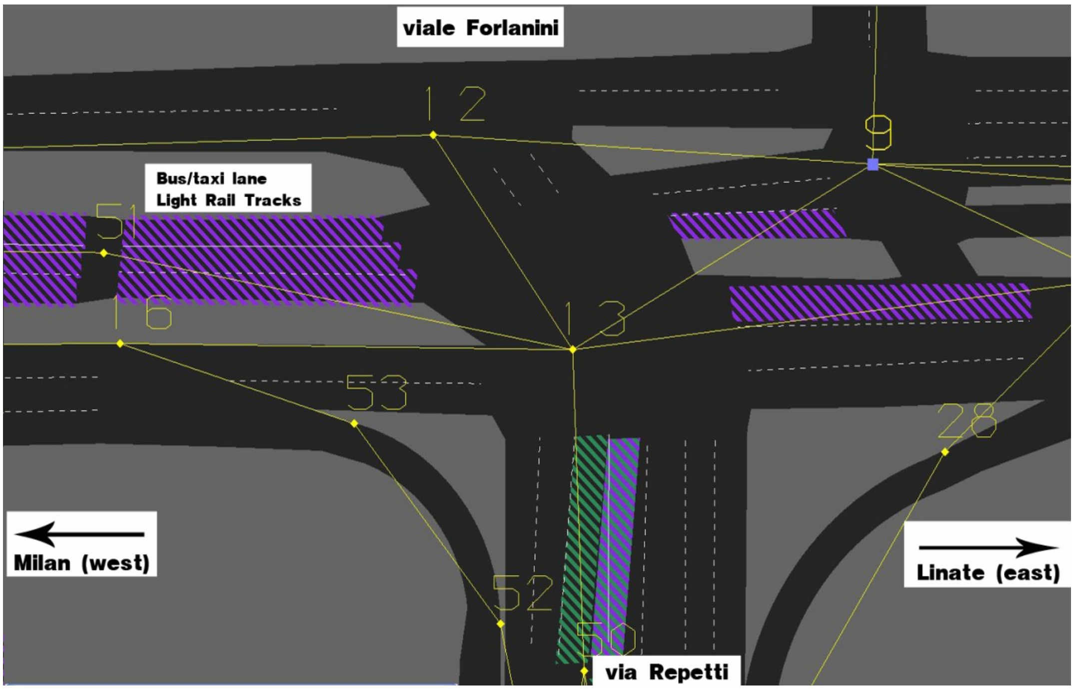


Figure 39: node 13 model

Because of the way the node model has been built, many unrealistic maneuvers are allowed. Right turn from via Repetti (path 50-13-17) must be barred because there is a reserved lane, the U turn with path 9-13-12 must be barred because there is the direct connection 9-12, the left turn from the eastbound carriageway of viale Forlanini and from the public transport reserved lanes to node 12 and the westbound carriageway are obviously forbidden, such as the U turn from viale Forlanini to its bus lanes in the middle of the road. In the end, the right turn from viale Forlanini to via Repetti must be barred through node 13 because of the reserved lane. The last detail of the "Junction" step is to switch the lane use of the bus lanes the way described before.

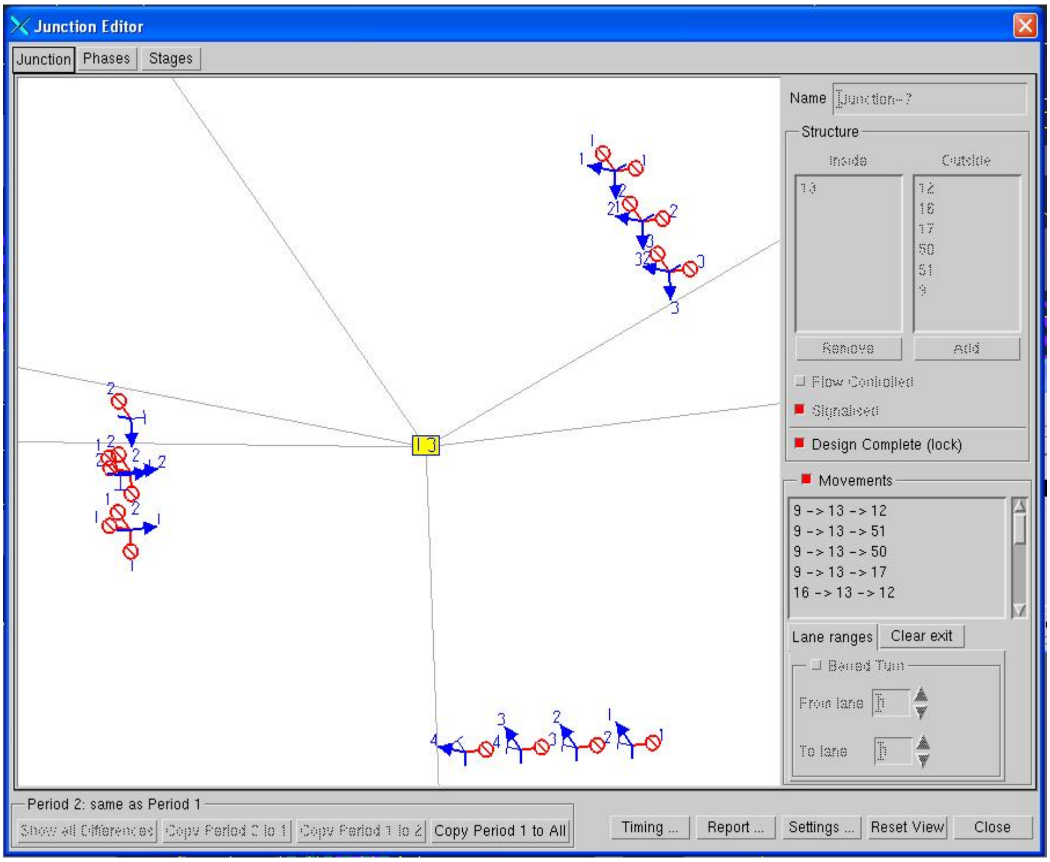


Figure 40: Junction editor with barred maneuvers and lane use of node 13

The “Phases” are similar to node 9, because of offsets between the greens each maneuver has its own phase. There are three “Stages”: the first (66 seconds of green) comprise maneuvers E (straight from the bus lanes with offset 10), A (offset 10 seconds) and D (straight from viale Forlanini with offset 13). The second stage comprise the left turns from the bus lanes of viale Forlanini eastbound to via Repetti and from all the lanes in via Repetti to viale Forlanini (phase B) with 20 seconds of green. Phase C, left turn from viale Forlanini westbound to via Repetti, moves during stage 3 with 19 seconds of green.

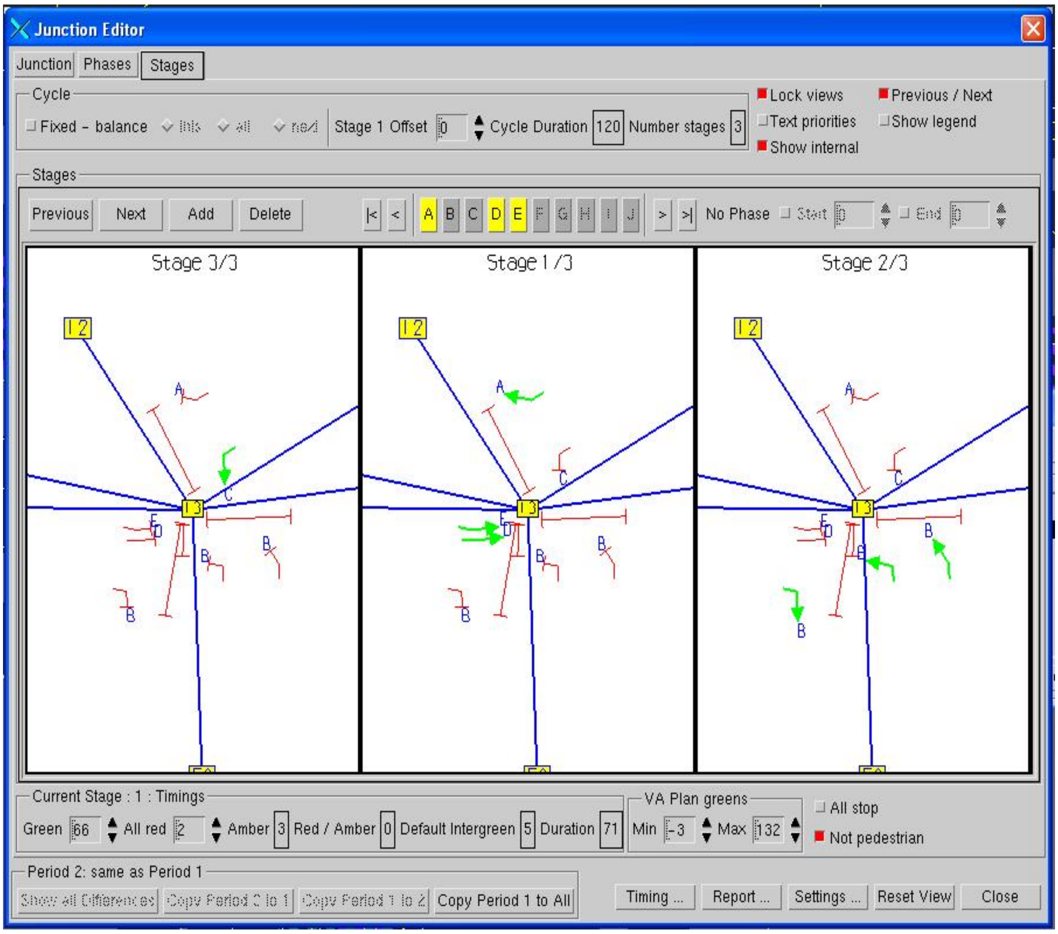


Figure 43: Stages for node 13

Here below the situation of traffic signals at each stage.

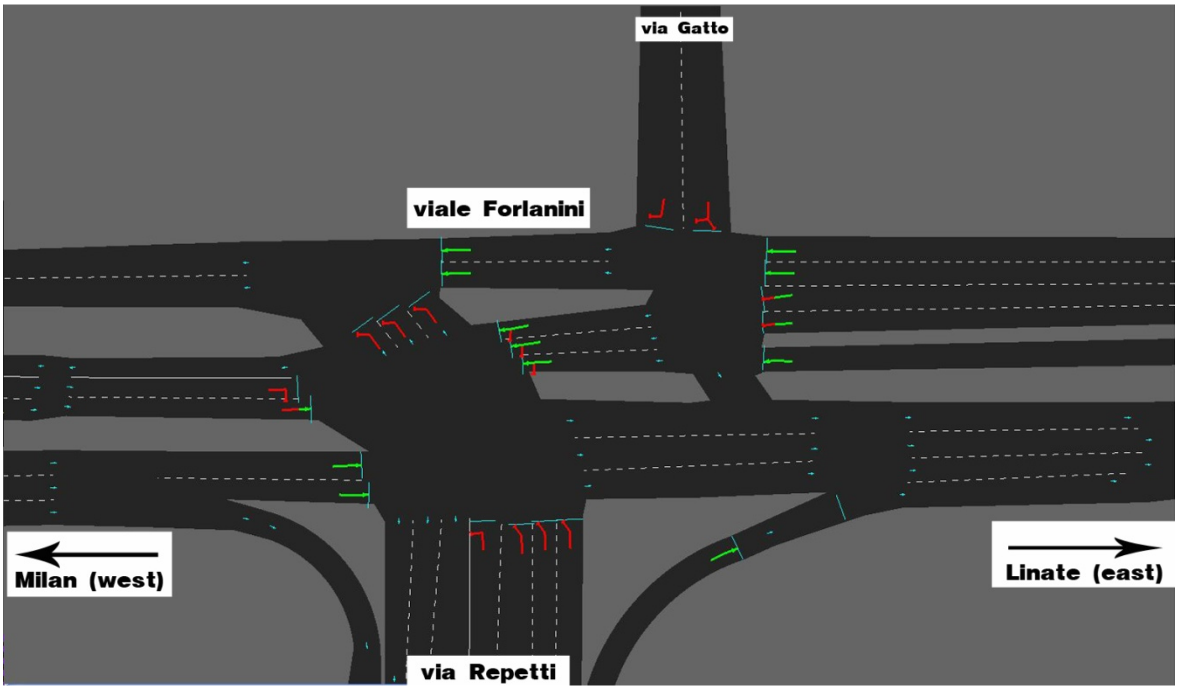


Figure 44: Forlanini - Repetti intersection stage 1

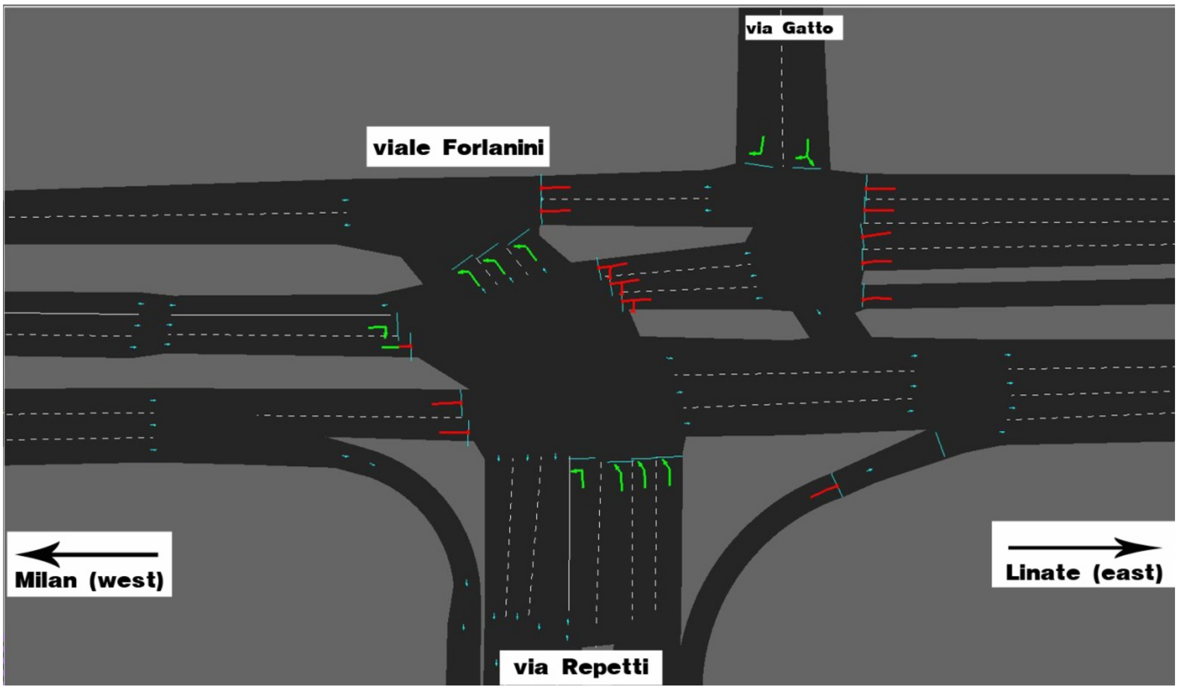


Figure 41: Forlanini - Repetti intersection stage 2

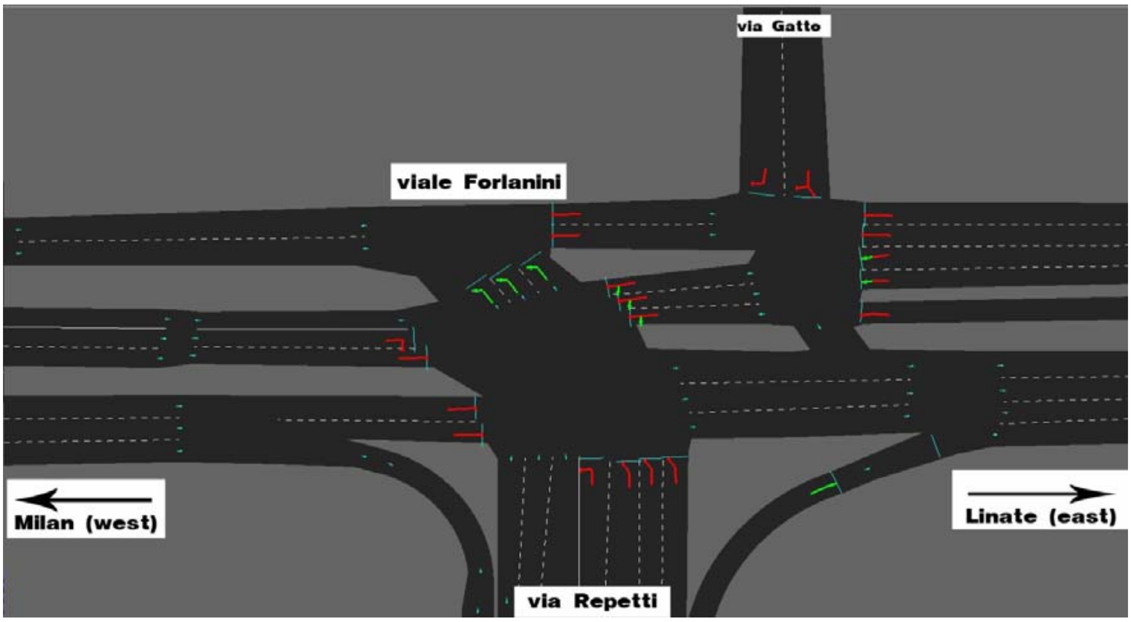


Figure 46: Forlanini - Repetti intersection stage 3

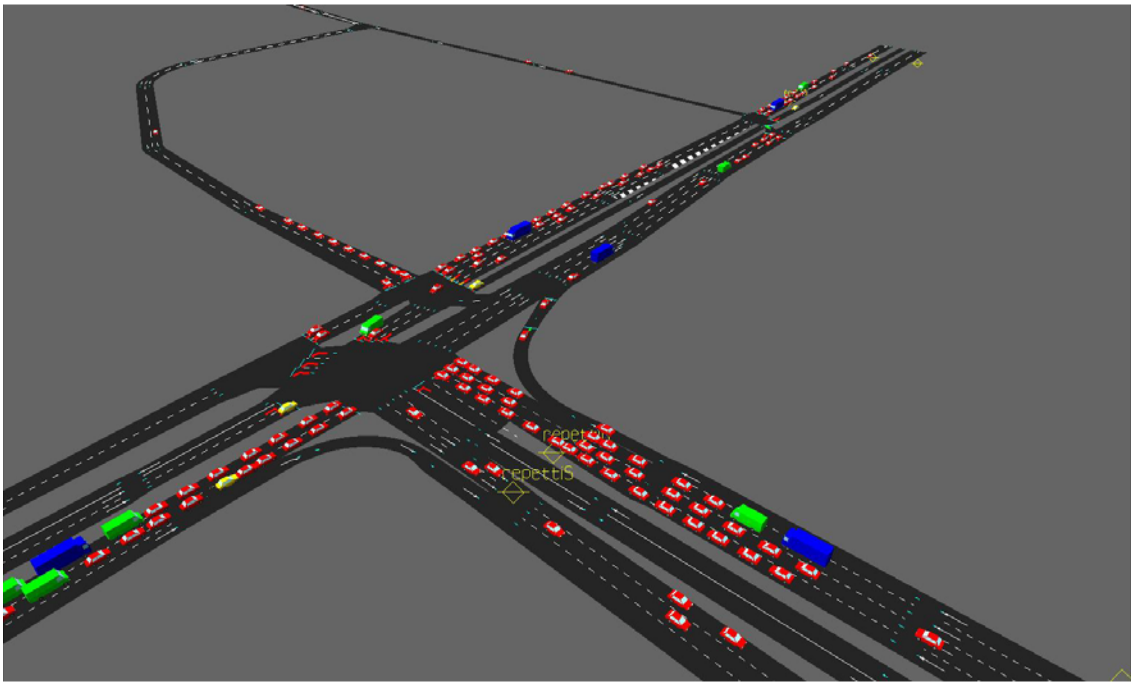


Figure 42: screenshot from the model run

4.3.3. The Project Scenarios

The goal of this model is to verify the impact on traffic of the Quartiere Forlanini Station building site. As discussed before, the station will be realized under viale Forlanini, a main road axis of Milan, with a traffic flow of 2000-2500 vehicles per hour per direction. The construction of the station will be divided into four stages with a top-down technique, that is digging a crater from the road surface down to the tunnels level.

On the first stage the eastbound platform and the south parts of the station will be built, closing the eastbound carriageway of viale Forlanini. Viale Forlanini will be reduced to two lanes per direction, all concentrated on the north side of the road. The westbound bus lane will be lost and it will not be possible to turn left on via Cavriana for drivers coming from Milan. The approach to the intersection will be kept with four lanes and a small priority lane for public transport and taxis.

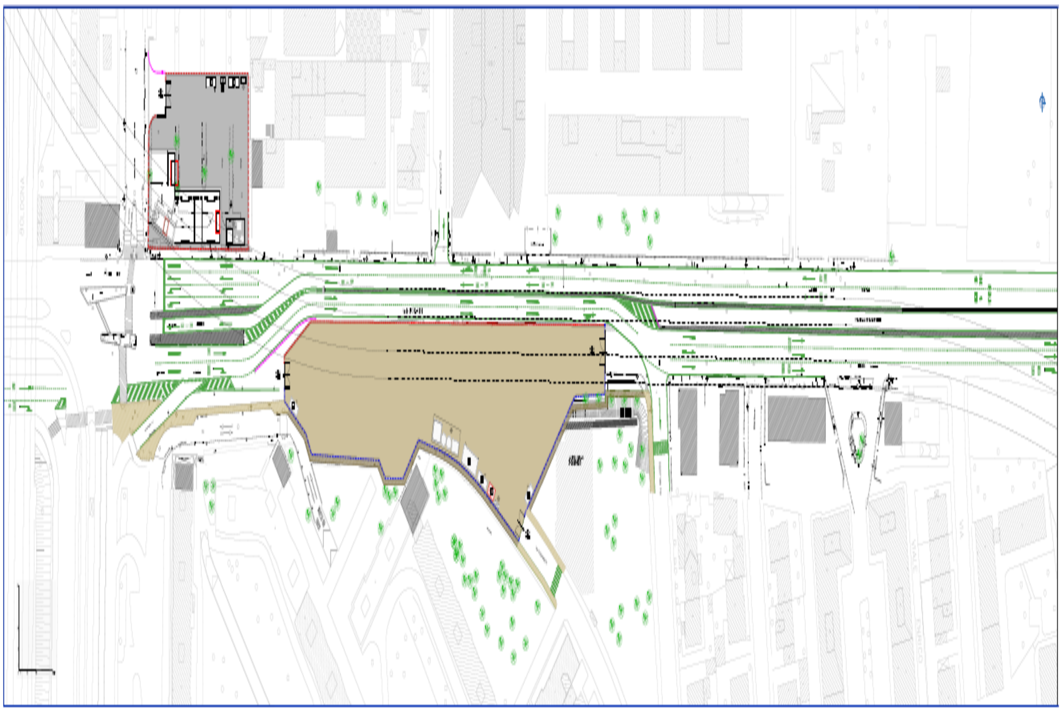


Figure 48: layout of phase 1 construction operations

During the second phase will be built the westbound platform and the north part of the station. The westbound carriageway of viale Forlanini will be closed due to the crater excavation and the situation will be exactly symmetric to phase one on the eastbound part of the road. Via Cavriana will be closed.

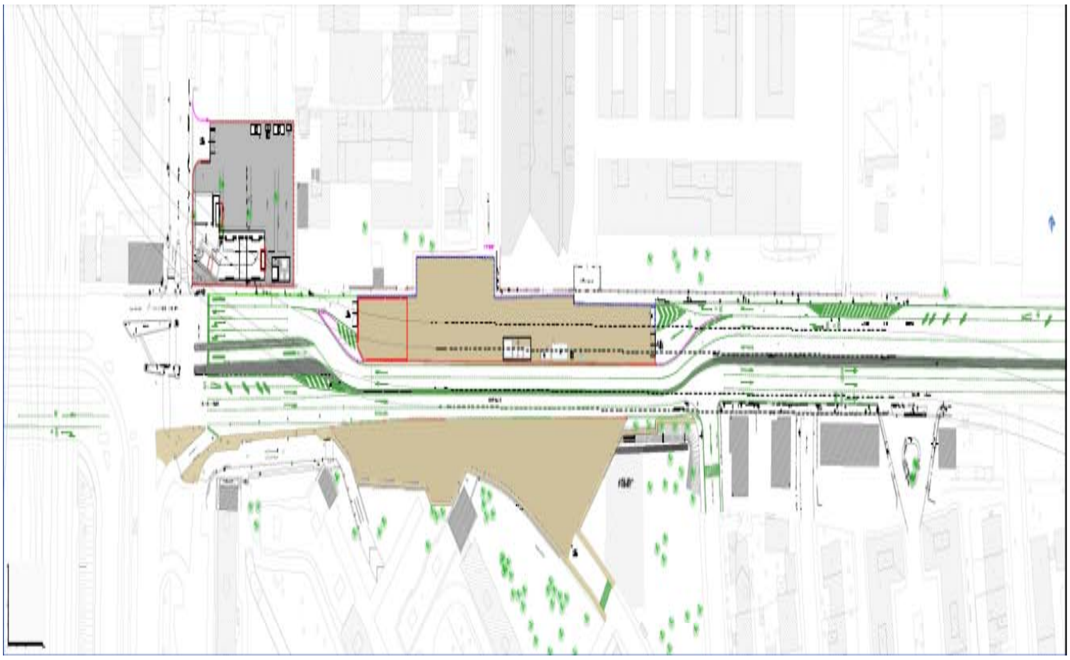


Figure 43: layout of phase 2 of construction operations

The third stage of the building site will be the one with the biggest impact on road layout. The central part of the station will be built, exactly in the middle of the road. The eastbound carriageway of viale Forlanini will be diverted on the area immediately south of the road previously occupied by the stage 1 building site, while the two westbound lanes will be separated and will enclose the building site itself. Via Cavriana will be open only for those coming from Linate Airport and accessible only from the external lane running on the north of the building site. The two lanes of viale Forlanini will then connect again on the intersection approach.

Figure 50: layout of phase 3 of construction operations

zThe fourth and final stage will be the one when stairs and surface connections of the station will be built. The building site will occupy the two sides of the road. Viale Forlanini will be reduced to two lanes per direction located in the central part of the road, exactly over the crater of the third stage. Via Cavriana will be open and accessible from Linate through the building site access road.

Figure 44: layout of phase 4 construction operations

For each stage we created a scenario modifying the road layout according to the drawings we received from the road engineers. We did not modify the traffic light cycle or other aspects of the model. To achieve precision, we superimposed on the model base (in general a technical map of the area) the drawings we received.

4.4. Model Calibration

Calibration is a fundamental part of any modeling process. “Calibration is the process of adjusting the parameters used in the various mathematical relationships within the model to reflect the data as well as is necessary to satisfy the model objectives” (DMRB Volume 12). Calibration means checking the model outputs against some real world information used in the model building process. The difference with Validation is that this second procedure is carried using real world data not used in the model development. Calibration is an iterative process, validation is not.

Because this case study represents a single intersection, there is no need for calibration. In fact, in this model there is no route choice: each zone is connected to any other zone by a single route. This is why, in this case, it would have been possible to assign traffic on itineraries instead of creating O/D matrices. Anyway, given the importance of model calibration, I will discuss this part of the work using another model I developed for the Sacelit-Italcementi real estate redevelopment in Senigallia, Italy, where there is the possibility of route choice.

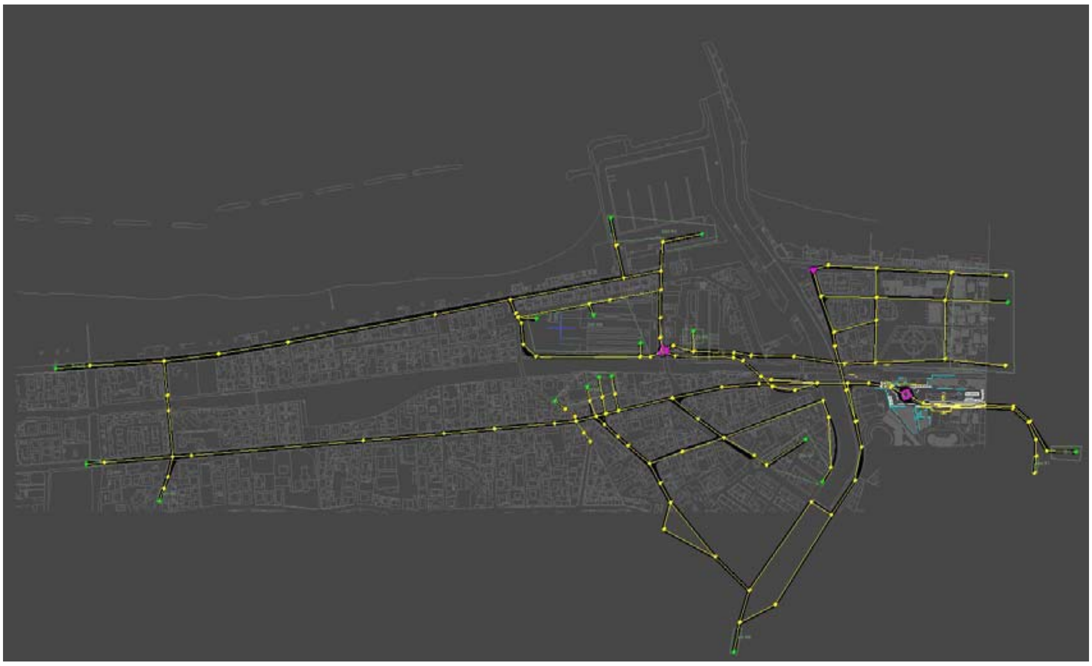
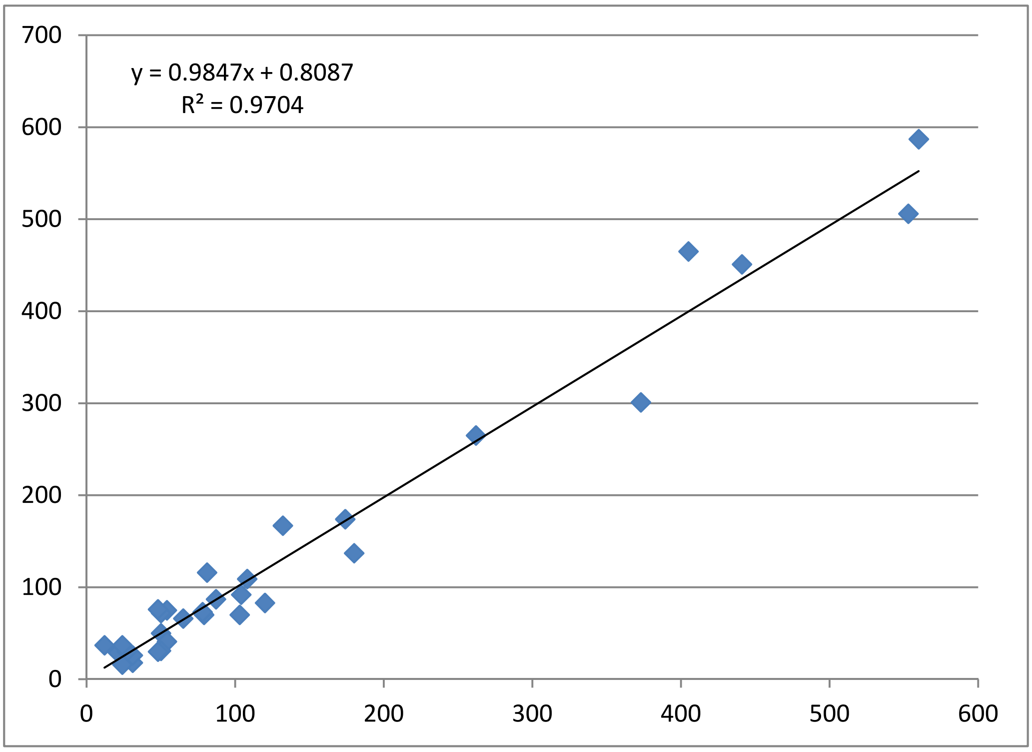


Figure 52: Senigallia model; green nodes are zone connectors (a zone can have more than one connector); pink nodes are roundabouts

To create this model we used 6 observation points. For each node, data on turn flows were available and from these data, following the Fourness process described in chapter 2.1.3, we obtained the necessary O/D matrices. We assigned the matrices to the model and verified the turn flow outputs against the available turn counts on the same maneuvers. This is essentially the calibration of the model.

Figure 45: traffic observation points in Senigallia

To measure whether a model is calibrated or not, two main parameters are available: the R^2 and the GEH. The first measures the variance from the scattering of simulated flow - observed flow pairs with respect to the average, represented by the bisector of the quadrant. The bisector has $R^2 = 1$. Here below an example of R^2 diagram for the Senigallia model. This model was quite good, with a $R^2 = 0.97$. The blue dots are the flow pairs (observed versus simulated); the equation of the best linear interpolation of points is represented as well.



Graph 1: the R^2 diagram for the Senigallia model

The other parameter, called GEH, defined by the British manual DMRB, Design Manual for Road and Bridges, as

$$GEH = \sqrt{\frac{(simulated\ flow - observed\ flow)^2}{(simulated\ flow - observed\ flow) * 0.5}}$$

The lower the GEH, the better the calibration of the model. The DMRB guidelines state that a model is calibrated if the GEH < 5 for the 80% of the maneuvers and never bigger than 10. To calculate the GEH is useful a spreadsheet like the one below, where the column “SIMULATED” represents the model output and “OBSERVED” the counts from the surveys.

OBSERVATION POINTS	MANEUVER	VEHICLES		ABSOLUTE DIFFERENCE	% DIFFERENCE	GEH	
		OBSERVED	SIMULATED				
1	A	553	506	47	9.29%	2.04	0
	B	21	29	-8	27.59%	1.60	0
	C	262	265	-3	1.13%	0.18	0
	D	441	451	-10	2.22%	0.47	0
	E	31	18	13	72.22%	2.63	0
	F	50	50	0	0.00%	0.00	0
	G	65	66	-1	1.52%	0.12	0
	H	87	87	0	0.00%	0.00	0
	I	81	116	-35	30.17%	3.53	0
	J	79	70	9	12.86%	1.04	0
	K	31	26	5	19.23%	0.94	0
2	A	132	167	-35	20.96%	2.86	0
	B	104	92	12	13.04%	1.21	0
	C	108	109	-1	0.92%	0.10	0
	D	54	41	13	31.71%	1.89	0
	E	54	75	-21	28.00%	2.61	0
	F	103	70	33	47.14%	3.55	0
3	A	560	587	-27	4.60%	1.13	0
	C	50	72	-22	30.56%	2.82	0
	G	120	83	37	44.58%	3.67	0
	H	27	30	-3	10.00%	0.56	0
	M	405	465	-60	12.90%	2.88	0
	N	50	31	19	61.29%	2.99	0
4	B	48	76	-28	36.84%	3.56	0
	D	24	37	-13	35.14%	2.35	0
	G	12	37	-25	67.57%	5.05	1
	I	180	137	43	31.39%	3.42	0
	K	48	30	18	60.00%	2.88	0
5	A	24	16	8	50.00%	1.79	0
	B	373	301	72	23.92%	3.92	0
	D	78	73	5	6.85%	0.58	0
	F	174	174	0	0.00%	0.00	0
						1.95	0.03

Chart 2: validation chart for the Senigallia model

If the model does not result calibrated, the model parameters must be modified. Chart 2 represents the second and last iteration of the calibration process. In the first iteration the 13% of maneuvers had a GEH over 5, and one a GEH = 9.38. Even if this was acceptable, we decided to improve the model and we obtained the results showed in chart 2.

This spreadsheet allows the user to simply copy and paste S-Paramics output in the right column and quickly verify the calibration of the new model runs.

4.5. Model Outputs

Both the models considered in this chapter, the Quartiere Forlanini subway station in Milan and the Sacelit redevelopment in Senigallia, are quite plain examples of TIS, Traffic Impact Studies. For this kind of projects, among the great variety of inputs a micro-simulation model can calculate, the ones of most common use are the number of simulated vehicles, the average travel time per vehicle, the total travel time, the total travel distance in the network, the average travel distance per vehicle and the current mean speed per vehicle.

Each scenario of the model (base, reference and project scenarios) is run several times to check the convergence of the model, i.e. the results of each run do not vary too much even if the all the random values of each run depend on a different seed value. If this is the case, the result of each model is represented by the run with the results closer to the average of all the runs. Once the “good” run has been decided, an output from S-Paramics shows the seed number, so that it can be input in the model to block the generation of different seed values and repeat, at each successive iteration, the exact same results.

The performance of the simulated projects is analyzed considering the confrontation between the base or the reference and the project scenario. The Base Scenario represents the current situation both in terms of demand and infrastructure supply. This scenario is used to calibrate the model with the observed data. The Reference Scenario implements the future demand, except for the traffic induced by new infrastructure or attracted/generated by new development, but including the growth in transport demand, all assigned on the current infrastructure. The Project Scenario considers the future demand, the completed future infrastructure demand with all the designed interventions and all the new traffic induced or generated/attracted by the new developments. In the Senigallia model the confrontation has been done between Reference and Project scenario. For the current case study, the building sites of the subway M4 in Milan, because there are no new developments or infrastructure, but just detours and possibly changes in the traffic signals plan to face the change in traffic flows, and the model simulates a construction period very close to the present, so the demand can be considered equal (the demand may actually decrease due to a re-routing effect caused by the knowledge of the users about the presence and the effect of the building sites), we analyzed the confrontation between the Base and the Project scenarios.

The main goal of this kind of studies is to verify that the proposed variations in road layout and regulations (or the increase of traffic due to the traffic induction effect of new infrastructure or attractors) do not significantly change the transport system performance. This evaluation can have different meanings: in the context of an infrastructural enhancement project, the system performance parameters must obviously improve. In the context of redevelopment on a consolidated urban area (such as the Senigallia case) or of long lasting building sites with impact on the transport network (such as the subway M4 building sites in Milan), it is perfectly acceptable a worsening of system performance parameters, as long as this is small.

The results for the Quartiere Forlanini station are presented here below. The most important parameter is the mean speed. In the Base Scenario the mean speed is 9.59 km/h (it is very small, but this is the model of a large intersection and the majority of vehicles is queued). The first phase of construction causes a decrease of speed to 8.18 km/h (-14.69%). The percentage variation is high because the speed values are small, but the absolute variation is only -0.72 km/h, too small to be noticed by the drivers. The speed reduction is caused by the reduction in the number of lanes and the consequent increase of congestion and queues, together with a reduction of the available space for queue build up that causes an increase in queue length.

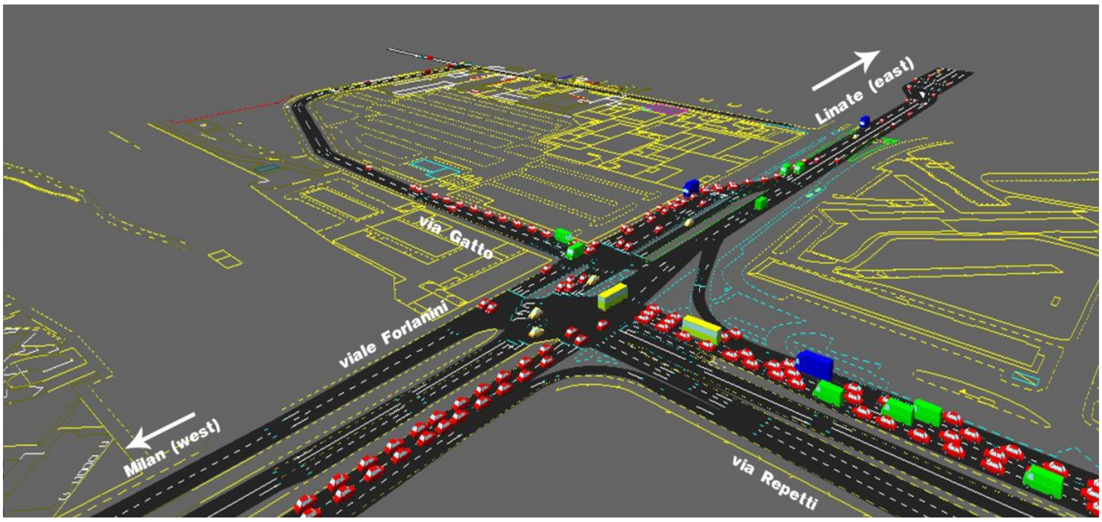


Figure 54: screenshot from phase 1 model

Scenario	PH_AM	Simulated Vehicles	Average travel time per vehicle (sec/veh)	Total travel time in the network (hrs)	Total travel distance (km)	Average travel distance per vehicle (km/veh)	Mean speed (km/h)
Base	07.30 - 08.30	6116	156.19	265.35	2545	0.42	9.59
Phase 1	07.30 - 08.30	5862	201.48	328.08	2685	0.46	8.18
Phase 1 vs Base	07:30 - 08:30	-254	45.29	62.73	139.50	0.04	-1.41
Phase 1 vs Base	07.30 - 08.30	-4.15%	29.00%	23.64%	5.48%	10.05%	-14.69%

Chart 3: results for phase 1 model

The phase 2 causes an increase of 0.48 km/h of speed, reaching 10.08 km/h. This improvement has two reasons: with respect to the Base scenario, the Cavriana intersection traffic light has been eliminated together with the related flow interruption and queue; also, the closure of via Cavriana causes the rerouteing of the traffic directed there, with a decrease of the demand at the intersection in the order of 400 veh/h during peak, thus causing a reduction of queue times and increase of speed.

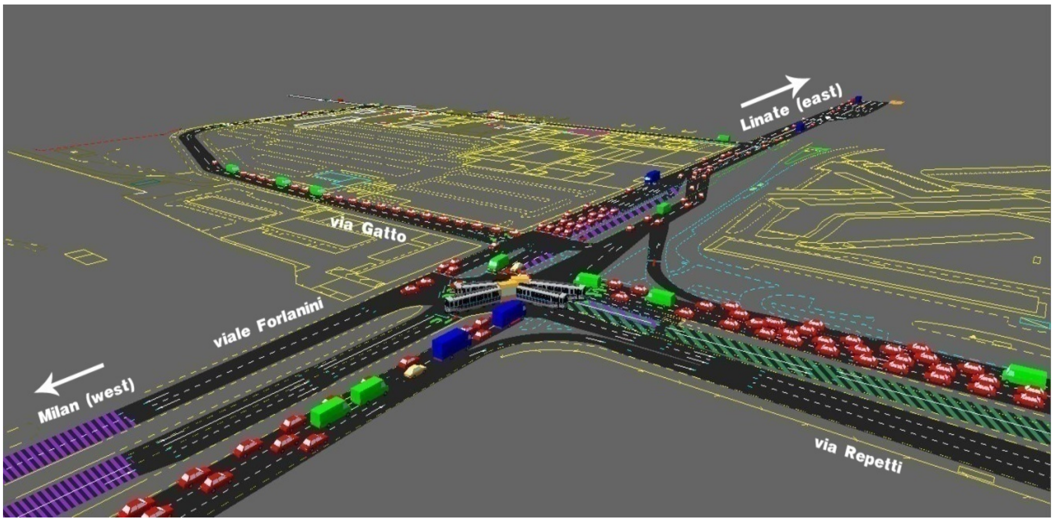


Figure 46: screenshot from phase 2 model

Scenario	PH_AM	Simulated Vehicles	Average travel time per vehicle (sec/veh)	Total travel time in the network (hrs)	Total travel distance (km)	Average travel distance per vehicle (km/veh)	Mean speed (km/h)
Base	07.30 - 08.30	6116	156.19	265.35	2545	0.42	9.59
Phase 2	07.30 - 08.30	5700	151.76	240.28	2569	0.45	10.69
Phase 2 vs Base	07:30 - 08:30	-416	-4.43	-25.07	23.55	0.03	1.10
Phase 2 vs Base	07.30 - 08.30	-6.80%	-2.84%	-9.45%	0.93%	8.29%	11.46 %

Chart 4: results for phase 2 model

During phase 3, the mean speed reaches 9.88 km/h, with an increase of 0.29 from the Base scenario speed (+3.05%). This is the combined effect of the speed increase due to the loss of the Cavriana traffic light and the speed decrease and longer queues because of a winding road layout.

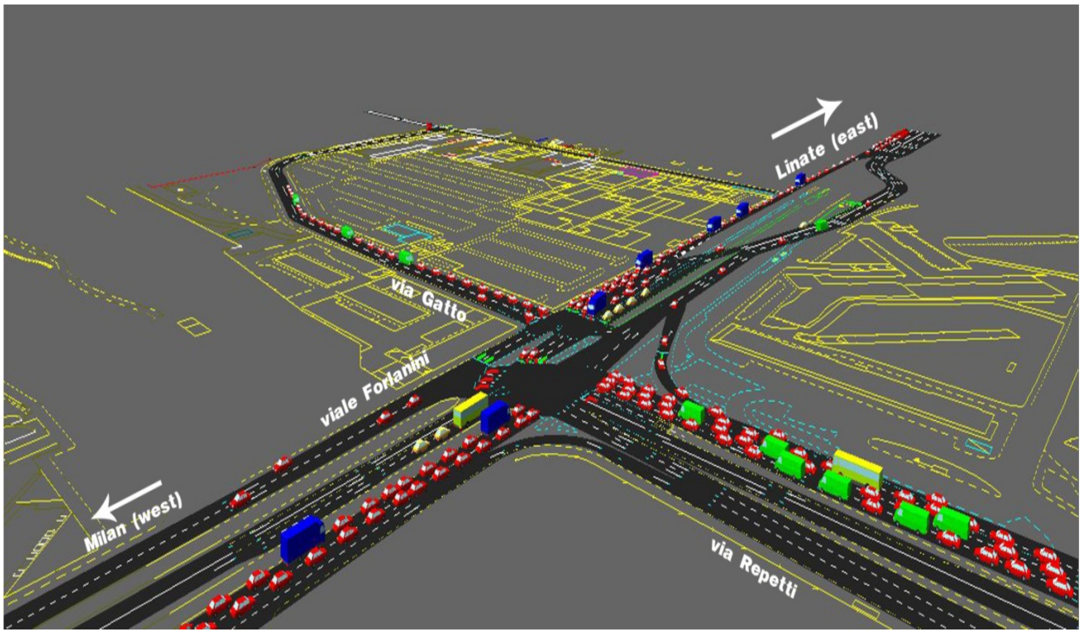


Figure 56: screenshot from phasee 3 model

Scenario	PH_AM	Simulated Vehicles	Average travel time per vehicle (sec/veh)	Total travel time in the network (hrs)	Total travel distance (km)	Average travel distance per vehicle (km/veh)	Mean speed (km/h)
Base	07.30 - 08.30	6116	156.19	265.35	2545	0.42	9.59
Phase 3	07.30 - 08.30	5969	163.12	270.46	2673	0.45	9.88
Phase 3 vs Base	07:30 - 08:30	-147	6.92	5.10	128.03	0.03	0.29
Phase 3 vs Base	07.30 - 08.30	-2.40%	4.43%	1.92%	5.03%	7.62%	3.05%

Chart 5: model results for phase 3

The fourth and last phase has the highest mean speed, 11.38 km/h (an increase of 1.79 km/h or +18.61% with respect to the base scenario). This is due to the elimination of the Cavriana intersection and the very linear road layout on both directions that improves the kinematic performance of vehicles and the increase the maximum speed of vehicles.

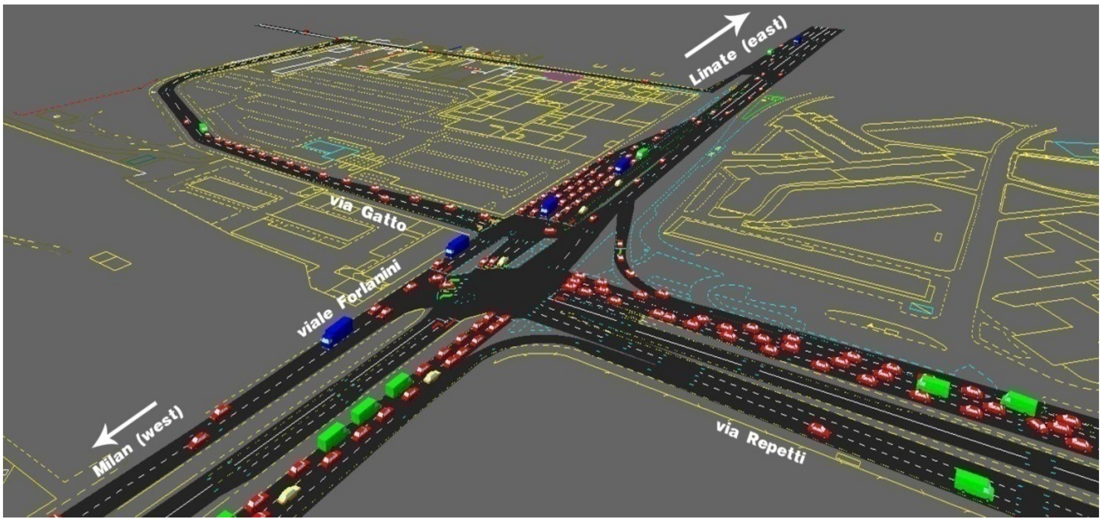


Figure 57: screenshot from phase 4 model

Scenario	PH_AM	Simulated Vehicles	Average travel time per vehicle (sec/veh)	Total travel time in the network (hrs)	Total travel distance (km)	Average travel distance per vehicle (km/veh)	Mean speed (km/h)
Base	07.30 - 08.30	6116	156.19	265.35	2545	0.42	9.59
Phase 4	07.30 - 08.30	5990	140.48	233.74	2659	0.44	11.38
Phase 4 vs Base	07:30 - 08:30	-126	-15.72	-31.62	113.99	0.03	1.79
Phase 4 vs Base	07.30 - 08.30	-2.06%	-10.06%	-11.92%	4.48%	6.68%	18.61 %

Chart 6: model results for phase 4

Overall, even if it is unlikely that the situation will improve with a building site reducing the road section, we can see from the model results that the proposed building site layouts will not cause a significant impact on the transport system. We have also to consider that on such a small scale model we did not considered the re-routing effect of the presence of the building sites.

It is of interest here to note that these layouts were only the last of a long series of proposals. It was actually the sixth layout design that we simulated. Here below some snapshots from the first simulation model. The first layout had two lanes on the eastbound direction and only one lane for the westbound traffic, with the other reserved for buses (initially, the municipality did not want to lose the reserved lane). Obviously, the capacity was way too reduced compared to the demand, causing big congestion problems and a traffic jam.

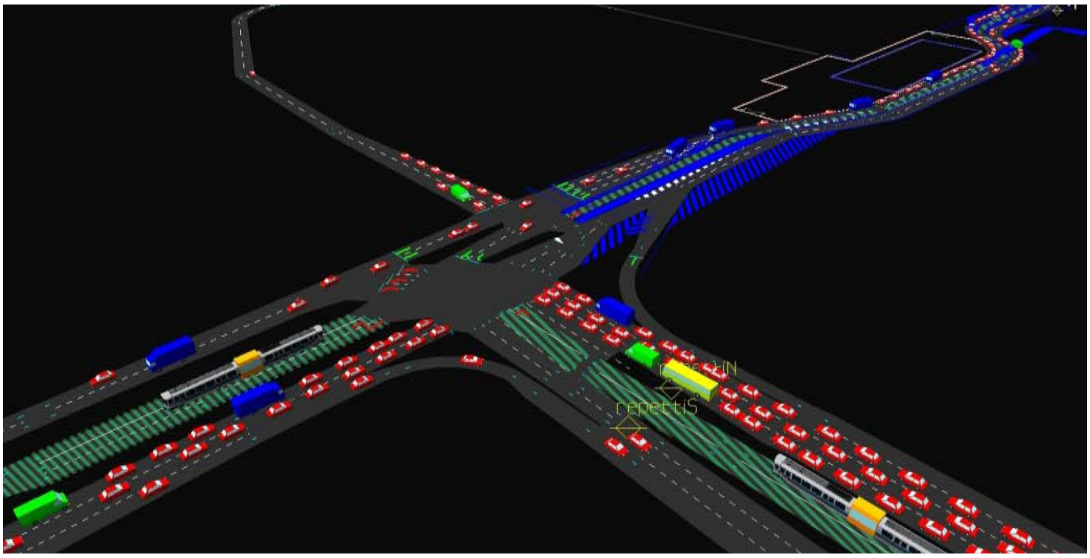


Figure 47: screenshot from layout 1 model

The second layout, with two lanes and a bus lane westbound, had a build up lane for left turn queues too short, and the queue blocked the other lanes causing congestion. Also, the road was too close to a neighboring residential building.

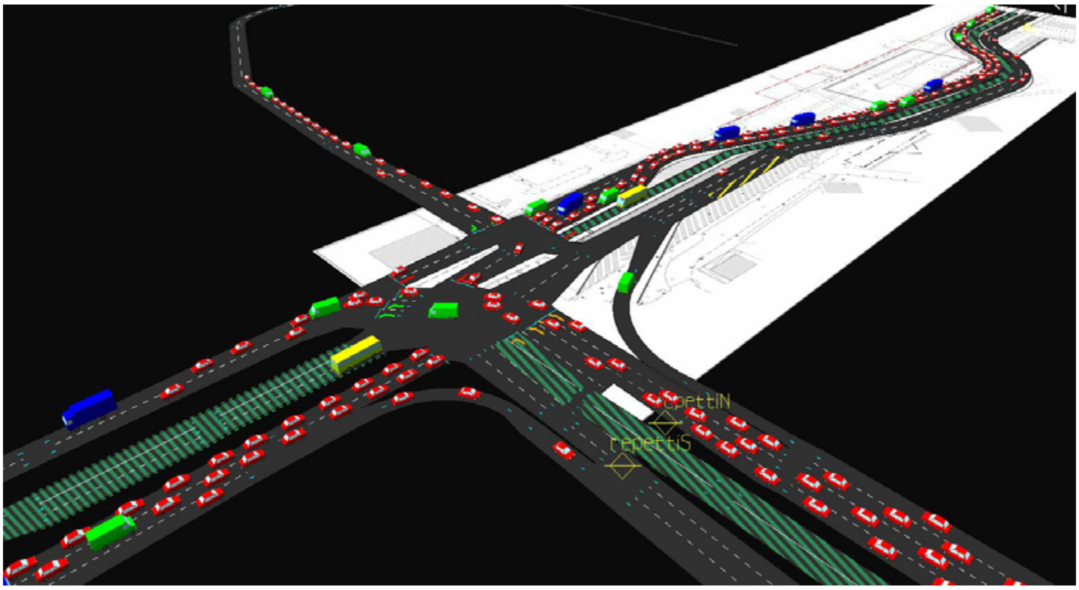


Figure 59: screenshot from layout 2 model

The third layout included an increase in the left turn queue build up lanes to a length of 50m. The situation improved with respect to the previous layouts, but the decrease in mean speed from the base scenario was still over 30%, clearly unacceptable. Again, the road was too close to the same residential building of the previous layout.

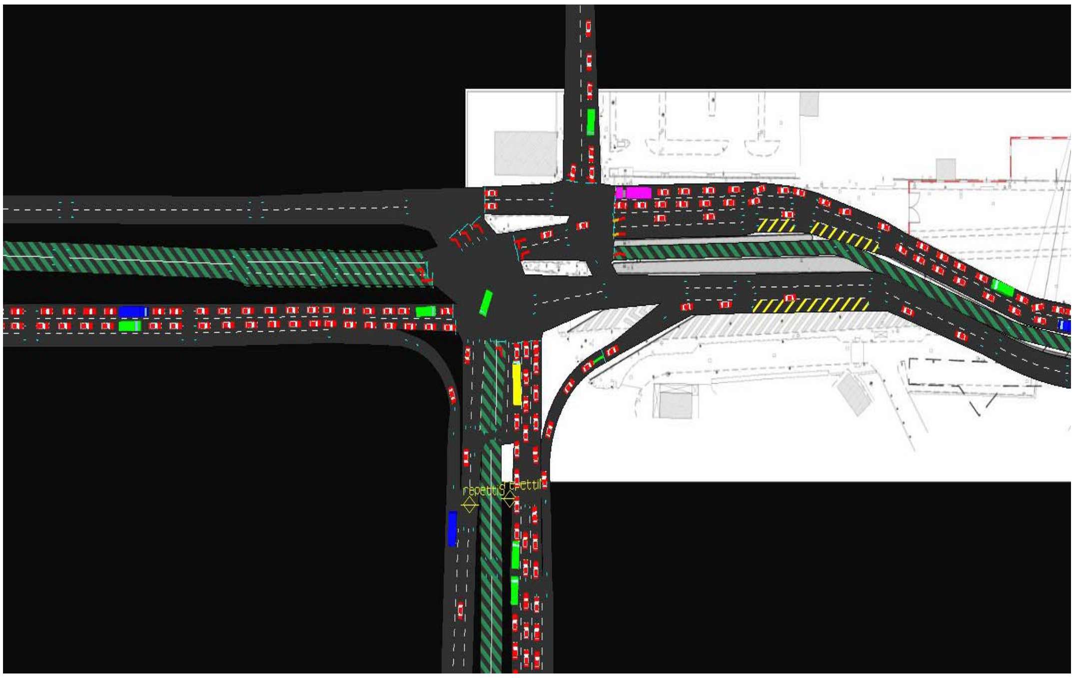


Figure 60: screenshot from layout 3 model

The final layout included in the third layout improvements in the traffic light regulations. This dramatically improved the situation, but the solution was deemed unacceptable again because of the proximity of the road to the usual residential building.

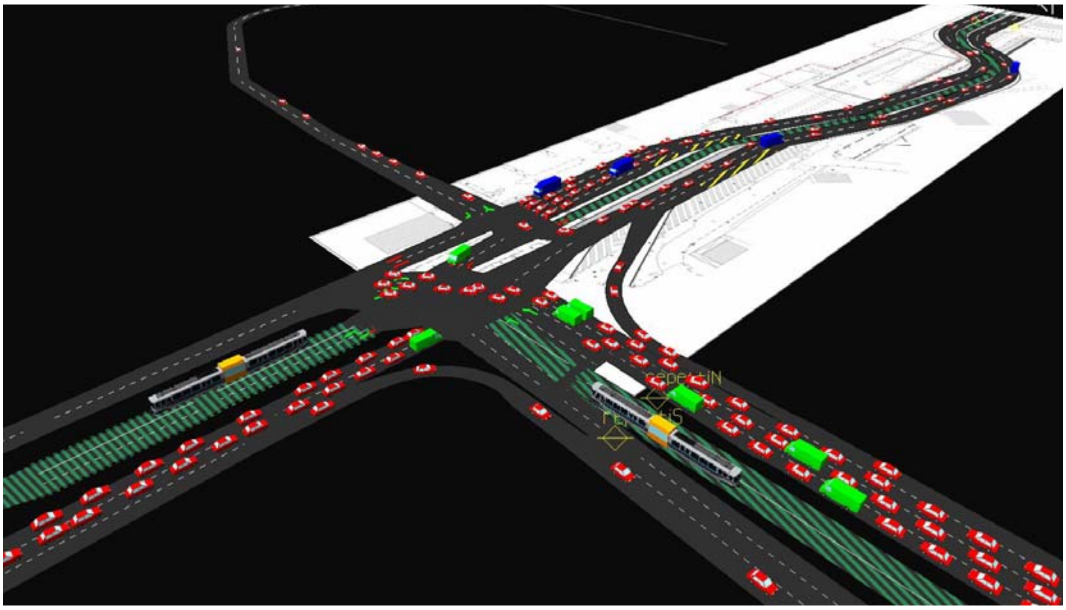


Figure 48: screenshot from layout 4 model

5. RESULTS

5.1. Overview

The present work analyzes the problems of micro-simulation, both on theoretical and professional level. At first, I described the transport modeling framework, from macro models to meso and micro models. In this section, I also analyzed how to gather, analyze and use social, demographic, economic and territorial (land use) data to calculate generation and attraction of territories and how to relate these data into a O/D matrices through the distribution and modal choice steps of the classical 4-step model for transport engineering. Evolutions of this model have been considered, such as the variable demand models where updates and iterations starting from the assignment results do not change only the modal split of each O/D movement, but also the demand between the zones, decreasing the theoretical demand for areas with low accessibility and increasing demand toward destinations with high accessibility.

As an information, I reported applications and studies of integration between meso and micro simulation that allow the study of wide area effects of punctual changes on the main parts of the network.

Demand modeling issues are not strictly speaking a problem of macro and micro simulation. Obviously, the different scales and purposes of the models require different data. Macro and meso models require as input data the trips generated by the relations between a set of attractors and generators of travel distributed on the territory. The relations depend on population composition, age, car ownership, wealth, public transport availability, infrastructure supply (presence of cycle paths, congestion charges, tolls, parking fees, etc.) and the utility assigned by users to each transport option available. The most modern demand models include also statistical data on willingness of the population to change their transport behavior or the presence of environmental concerns that may lead to more sustainable choices even if not the most convenient in terms of time/cost. The O/D relations can be modified as well by congestion and accessibility issues.

5.2. The Simulation Process

The simulation process itself, no matter if macro or micro, deals with the assignment. Macro models consider demand and infrastructure supply as if they have constant characteristics during the simulation period. The characteristics are modified at each iteration considering the results of the previous calculations. Macro models perform what is called a static assignment. Micro simulators, on the other side, update the road and traffic conditions (or, more precisely, the users' perception of road and traffic conditions) with very short time steps (around 1 second) to simulate the adaptive behavior of drivers. Micro models perform dynamic assignment, that is the simulation period is divided into many smaller time periods during which the characteristics of the transport system change according to congestion, and the route of freshly released vehicles is calculated according to the current traffic conditions.

If this is a way more realistic representation of reality, this requires a way larger amount of data with respect to macro models. Macro models require the demand data as outlined before, number of lanes and capacity of links and the reductions of capacity, with respect to free flow conditions, due to traffic signals and other intersections. Micro models require , to achieve realistic results, a detailed reconstruction of road geometry and layout, number and use of lanes, signal plans, difference between give way and stop signals and, for each link, cost factors, maximum speed, lane width, etc.

The problem of the amount of data has not been solved yet even in software like SUMO, that brings micro simulation very close to strategic predictive models. For cities with thousands of links and zones, the amount of data required is huge, and their manipulation into a demand model is a massive task. The problem here is if the effort is justified by the improvement of results with respect to the traditional approach based on macro and micro models. Many experts do not consider this effort worth the results and state that this approach is bringing micro simulation too far from its application domain. Anyway, many cities, such as Stockholm, and even entire regions in the Netherlands, created large scale micro simulation models or integrated micro-meso models representing the whole urban areas.

Large scale micro models require necessarily less detail data to avoid excessive effort compared to result. Again, the problem is whether the decrease of detail in micro simulation, a procedure created to analyze in detail local traffic problems, makes sense or it just bring micro-simulation out of its application domain. The argument is still open to discussion.

When it comes to demand, less data, or at least, easier to collect demand data are necessary: in the end, just flows on the considered links, or on the boundary of the study area, are required. These data can be the outputs of a macro model or, more often, can come from traffic counts and surveys. Matrix manipulation techniques, such as the Fourness algorithm, are then used to create and correct the O/D matrix. The flows must be divided into different vehicle categories through counts and/or surveys. All the socio-economic implications of transport demand generation do not interest micro simulation. In fact, micro simulation is not a predictive tool and can only consider the short term effect of variations in the transport system. Micro simulation cannot analyze wide area re-routing effect or changes in the modal split.

Another aspect present in micro-simulation is the characterization of users. In fact, macro models model all the users with same general parameters. Micro-simulators determine the users' behavior through a set of editable characteristics, most of them defined through statistical distributions. Also, micro simulators need to define the kinematics of vehicles and their interactions. Many models exist for the purpose. They can be car-following models that control acceleration, cruising and deceleration of vehicles, merging models to control the lateral behavior of users when merging or overtaking and finally the physical models that control the turning of vehicles, especially in tight turns maneuvers at intersections. In chapter 2 I analyzed the models used in S-Paramics, the software I used to develop the case study model. SUMO has the

possibility to choose among many models. In chapter 3 I mentioned the models used in VISSIM and the differences with the S-Paramics approach. Good kinematic and physical models are fundamental to achieve a correct simulation, not only at a graphical level, but also in more important calculations such as delays at intersections and queues. Because the detail of micro modeling and the detail of expected outputs, a detailed representation of vehicle movements, speed and interactions is obviously fundamental because even small inaccuracies, repeated thousands of times, can lead to large errors.
Finally, the outputs. A wide range of outputs can be obtained from a micro simulation. Incidents, emissions, noise pollution, delay for public transport or for random vehicles, queues, turn counts, car parks occupancy and more traditional data, such as mean speed, link or network delays, vehicle delays, travel times and travel distances. Results can be obtained for the whole model, for single links, single vehicles or areas of the model. Because micro simulation, in the private sector, is mainly used for Traffic Impact Studies of new developments or to verify important but temporary changes to the road network (such as the subway stations building sites of the case study), what is of interest is the second set of parameters (total travel time, average travel time per vehicle, total travel distance, average travel distance per vehicle and vehicle mean speed) to verify that the proposed solution improve the traffic flows, or at least does not make the situation worse with respect to user perception. In general, this means that if, for example, the mean speed for the network is 30 km/h, it is an acceptable situation one that guarantees a mean

speed of at least 27-28 km/h (so that the user does not realizes the difference) given the new demand or the capacity reduction.

5.3. The uses of micro-simulation

To sum up, micro simulation models are used today to verify punctual situation, such as changes in complex intersection regulations, layout or traffic signal plans (for easier intersections static software like SIDRA is used). Micro models are also used for Traffic Impact Studies of a variety of situations, from new developments or re-developments to temporary but large variation of road layout and/or capacity, such as long lasting building sites on the road. A standalone micro model cannot analyze mobility patterns or modal shifts due to variations in transport supply such as improvement of public transport, creation of cycle paths, variation in accessibility of different areas or new road layouts. Researches in this direction are being carried out integrating micro models into larger models, such as meso models, to analyze the wide area effects, including re-routing, of punctual modifications. The integration of micro models into public transport demand models can provide precise output in term of travel time and speed for buses, used as inputs into demand generation and modal choice iterations, whose outputs are then used to simulate the effect of modal shifts and demand variations on circulation, creating new micro simulation outputs (speed, travel time) to update the demand model, in an iterative process.

6. REFERENCES

S. Saiyed, *An Assessment of Arterial Network using Macro and Micro Simulation Models*, Calgary (Canada), 17th Annual International Emme/2 Users' Conference, 2003

Systematica S.r.l., *Stazione Forlanini Quartiere – Analisi di funzionalità: scenario di cantiere*, Milan (Italy), 2012

AMAT, *Accordo di Programma per la trasformazione urbanistica in variante al PRG vigente delle aree ferroviarie dismesse e in dismissione*, Milan (Italy), 2009

G. Duncan, D. McArthur, *Paramics Technical Report: Car-Following, Lane Changing and Junction modeling*, Edinburgh (UK), 1997

P. S. Goodman, *The Prediction of Urban Traffic Noise in Urban Areas*, Leeds (UK), The University of Leeds – Institute for Transport Studies, 2001

R. Horowitz, *Development of Integrated Meso/Microscale Traffic Simulation Software for Testing Fault Detection and Handling in AHS*, Berkeley (USA), University of California Berkeley, 2004

R. A. Harsløf Holst, *Macro-, Meso- and Micro simulation – A comparison of three DTA softwares*, Copenhagen (Denmark), Danish Technical University, 2007

W. Burghout, *Traffic Simulation Models, Part 2: hybrid meso-micro*, Stockholm (Sweden), CTR, 2007

TFL, *Micro-Simulation modeling guidance note for TFL,* London (UK), Transport For London, 2003

G. Kotusewski, K. A. Hawick, *A review of traffic simulation software*, Auckland (New Zealand), Massey University, 2009

C. Robinson, G. Bell, M. Brewer, S. Canning, K. Lumsden, *Transport Model for Scotland – Demand Model Development*, Edinburgh (UK), Transport Scotland, 2007

Michael Behrisch, Laura Bieker, Jakob Erdmann and Daniel Krajzewicz, *SUMO - Simulation of Urban MObility: An Overview*, SIMUL 2011, The Third International Conference on Advances in System Simulation, 2011

P. Ryder, *The Use and Application of Micro-Simulation Traffic Models in Design Manual For Road and Bridges*, London (UK), Department of Transportation, 2001

S. Wood, *Modeling Variable Demand Using Micro-Simulation*, Edinburgh (UK), Sias Ltd, 2011

ATM, *Analisi della domanda di trasporto e valutazione del carico per la linea metropolitana 4 di Milano*, Milan (Italy), ATM, 2010

Systematica S.r.l., *Riqualificazione dell'Area Sacelit-Italcementi, Sistema Infrastrutturale di Accessibilità e*

Viabilità, Studio di Traffico, Milan (Italy), Systematica S.r.l., 2012