1. INTRODUCTION

Traffic assignment models based on user equilibrium approaches are one of the most widely used tools in transportation planning analysis. The modeling hypotheses, soundly based on the well established concept of equilibrium, lead to nice mathematical models for which there are efficient algorithms that provide solutions in terms of the expected flows on network links, Florian and Hearn, 1995. Modeled flows offer a static average view of the expected use of the road infrastructure under the modeling hypothesis. This information has usually been sufficient for planning decisions, namely at regional and metropolitan levels, as far as the optimization algorithms, based on user equilibrium principle, have a quite efficient computing performance even for very large networks.

However, the evolution of advanced technologies and their application to modern traffic management systems require in most cases a dynamic view complementing the static estimates provided by the assignment tools. The planned infrastructure is probably sufficient for average demand, but time-varying traffic flows, i.e. at peak periods, combined with the influence of road geometry, can produce undesired congestion that cannot be forecasted or analyzed with the static tools, as far as models based on static network equilibrium cannot capture the time dependent traffic phenomena as queue spillbacks or the time evolution of congestion. This is a clear case for changing in the analysis methodology.

The lack of an algorithmic implementation of dynamic equilibrium models computationally efficient for medium or large networks, and the progress made by microscopic simulators in the late 90’s led practitioners to think of the convenience of interfacing a macroscopic approach for transportation modeling based on user equilibrium traffic assignment models, and a microscopic approach, as implemented in microscopic traffic simulation models, as a way of overcoming these limitations, and in recent years commercial software interfacing EMME/2 and AIMSUN, VISUM and VISSIM, or TRIPS and DYNASIM has put to work in practice these ideas.

However, as far as microscopic models are data intensive, the fact that the calibration of models for large networks is not usually an easy task, and the failure to achieve in some cases a consistent dynamic assignment led researchers to consider a different modeling paradigm, the so called mesoscopic models, to fill the gap between the upper level aggregated approach used in the macroscopic models and the lower level fully disaggregated approach of the microscopic models. Mesoscopic models, describing in a simplified way the flow dynamics and implementing heuristically a dynamic equilibrium, can computationally succeed in the analysis of large networks. DYNASMA, Jayakrishnam et al. 1994, DYNAMIT, Ben-Akiva 2002, and more recently DTASQ, Florian et al. 2001, and Mahut et al 2004, are good examples of such approach.

Although the criticisms on the ability of microscopic modeling to achieve a proper dynamic assignment have been overcome, proving that a proper implementation of link costs and route choice can achieve the expected results, Barceló and Casas 2004, and the improvement in hardware and software have speeded up dramatically the computational performance of microscopic simulators, the fact is that each approach plays, and very likely will pay for a long time, a well defined role in transportation analysis as far as each one has unique characteristics to answer appropriately certain questions that the other cannot answer that well. Consequently the discussion is not whether one approach is better or more appropriate than other, or if there is a unique approach that can replace satisfactorily all others, but which is the most appropriate use of each approach and how can then work together in a common framework, if possible, generalizing the former macro-micro interfaces to a combination of macro, meso and micro, in a new methodological framework that could be considered as an extension of the traditional four steps planning model into a six steps planning model, Jayakrishnam 2005, in which the four steps of the usual static,
strategic long term planning, would find a follow up in term of a dynamic medium term planning, and short term dynamic tactical analysis or, in other words, the corresponding macro, meso and micro levels.

A proper interfacing of two modeling approaches, macro and micro, based on so different principles raises a set of methodological questions, Montero et al. 2001. A microscopic route based simulation in which vehicles follow time changing paths from origins to destinations can be used to get a deeper insight of the performance of a planned infrastructure, complementing in that way the estimated of a classical planning exercise based only on traffic assignment. A proper interface between the two models requires that both network representations at macroscopic and microscopic levels respectively are consistent, and that the same Origin-Destination matrix is used in both approaches. A way of satisfying accurately this requirement is to use a model building tool that automatically translates the network representation at disaggregate level, that of the microscopic view, into the aggregate level of the macroscopic view. Assuming that network consistency is appropriately solved there still remains what is likely the most critical consistency problem: how to synchronize changes in the models, that is how a change at macro level is translated into the meso or micro level and reciprocally. Another key methodological questions are whether the time dependent paths used in the microscopic simulation can be interpreted in terms of a dynamic user equilibrium or not; or how these time dependents paths at micro level correspond to time dependent paths at meso level, Wilco Burhout et al. 2005, discuses these consistency problems of making a meso and a micro approach work together. This paper presents an integrated framework to support a transport analysis methodology that combines the three approaches, overcoming these problems. From a modeling point of view there are two main issues to achieve these objectives:

- Computer modeling issues: an efficient integration of transport analysis tools requires an specific computer modeling approach to overcome the inefficiencies and limitations of transport models based on different network representations communicating by means of file exchanges
- Traffic modeling issues concerning the modeling paradigms to be used at each level and their compatibility.

2. A COMPUTER MODELING FRAMEWORK TO INTEGRATE MACRO, MESO AND MICRO APPROACHES

Macro, meso and micro approaches to traffic modeling although sharing some common concepts are based on different types of network representations. All them share in common the demand model in terms of an Origin-Destination Matrix whose entries represent the number of trips from the selected origin to the chosen destination, for a given time period and trip purpose; origins and destinations being represented in the model in terms of artificial nodes, or centroids, where traffic flows are generated and sunk. The differences in demand representation lay in that, for macro approaches usually based on static user equilibrium, the demand matrix is unique for all the time horizon considered, while in the meso and micro approaches the demand matrix is usually split in a set of matrices for smaller time period spanning the whole temporal horizon, to approximate better the time varying traffic demand. Centroids are connected to the model of the physical network in terms of dummy arcs or connectors, how connectors are connected to links or nodes of the network model determine the demand behavior, therefore a first task to ensure the consistency of the demand representation in the three approaches calls for ensuring that the centroids and their connections are the same at the three levels.

Macro and meso approaches are based respectively on link node and extended link node network representations, with nodes modeling intersections and links modeling the transportation infrastructure, while microscopic modeling to traffic networks requires an explicit and detailed modeling of the road sections, intersections, roundabouts and so on. Road sections at macro and meso levels, are therefore represented in terms of abstract oriented arcs, whose characteristics are defined in terms of numerical attributes, i.e. capacity, number of lanes, associated volume-delay function, or speed density relationship in the extended link node, etc; while at microscopic level lanes must have a width, and other additional numerical attributes are necessary, as for example the speed limits on the section, although macro and meso, or meso and micro, can share some of the numerical attributes. At macro level nodes representing intersections include a detailed definition of the allowed turnings, but no information on signal setting at signalized intersections, or specific give way or stop rules at unsignalized intersections as far as they do
not account for signal control, while meso and micro approaches need all the information concerning phasing schemes and time settings at controlled intersections as they deal explicitly with signal control, as well as information on give ways and stops at unsignalized intersections to model traffic behavior in a suitable way in these cases. That means, usually, that when moving form the upper aggregated level of network representation of macro approaches to the lower level of meso and micro there is missing information that should be additionally provided to ensure the model consistency at the corresponding lower level, and when moving in the opposite direction, from the fully disaggregated lower microscopic to the upper aggregated levels meso or macro, the appropriate aggregation rules must be defines to ensure a unique way and keep the models consistent.

The ideal situation is that depicted in Figure 1, where starting at the macro level with a very large model of a region or a metropolitan area the analyst, after conducting a transport planning analysis windows into a large sub area (possibly the whole area spanned by the model) to get a deeper insight accounting for the dynamic effects of time dependencies of traffic phenomena.

Figure 1: From macro to meso and micro

The dynamic analysis may reveal potential conflicts in smaller sub areas requiring a further detailed analysis, namely when one should account for ITS applications, like adaptive control strategies, for which microscopic simulation is the suitable model.
How to communicate and put the three approaches to work together in an efficient way? When the modeling approaches have been developed independently, implemented in independent software packages the exchange of information between the three levels based on file exchange is the only but not the most efficient way, and requires a complex overhead when also time dependent paths of vehicles traveling across the network from origins to destinations must be taken into account as in the case of hybridizing meso micro approaches, Burghout 2004.

An efficient answer to these questions must be based in a flexible combination of computer modeling and transport modeling, which is an implementation of transport models supported by advanced software architecture. Figure 2 depicts a conceptual diagram for such architecture, as implemented in AIMSUN NG, TSS 2005. The system is conceived as an integrated environment enabling the direct exchange of information among various transport analysis tools.

![Figure 2: Conceptual architecture of AIMSUN NG](image)

The key component is an extensible object model common to all applications that share a unique Model Database. Transport Planning, mesoscopic simulator and microscopic simulator, among other applications exchange the information directly in this way, each one using the information that needs and modifying the stored information accordingly, all the data have a common storage and the type of network representation depends only on which subset of data is used. This structure is schematically represented in Figure 3. The object model for the Macro level contains attributes exclusive of the macro network.
representation and attributes which are shared with the Meso level, and similarly, Micro level shares some of the attributes with meso and macro, but has other exclusive of the macro level, but all them are stored in an unique model database.

This allows simultaneous network level representations as those depicted in figure 4. On the left there is the view of the link node representation of a network, as required by a static user equilibrium assignment, the window shows part of the attributes of a selected link, some of them representing data for a microsimulation, or for the mesoscopic, as the Jam Density, and others, as the volume-delay function included in the graphic, to be used in the user equilibrium assignment at macro level.

On the right part of the figure depicts the microsimulation mode of the same network, the window shows the details of the turning movements and timing allocated to one of the phases of the selected signalized intersection.

This multilevel network approach made possible by the computer modeling of the transport system allows to conduct the static equilibrium assignment, the mesoscopic simulation or the microscopic simulation simultaneously executing each model on its window. Figure 5 depicts the case in which on the left window has been executed the user equilibrium assignment, link colors visualize the expect level of service after the assignment, the central window visualizes similar results for a given time interval from a mesoscopic simulation, and on the right window is being executed a microscopic simulation and the corresponding animated view visualizes the individual vehicles, queues at intersections, etc.

3. COMBINING MACRO WITH MESO OR MICRO: WINDOWING INTO SUBAREAS

As it has already been mentioned, one of the reasons to combine the various approaches is to zoom into a subarea of a larger area in which a problem has been detected in the analysis at the higher level, i.e. after conducting a transport planning study conflicting points are identified that require a deeper and more detailed analysis of dynamic type. Depending on the characteristics of the identified problem a meso or a micro analysis should proceed.

The desired methodological process, whose conceptual diagram is depicted in Figure 6, is illustrated in Figure 7. The upper level corresponds to a macro model, a graphic window defines the subarea of interest, which model, meso or micro, depending of the objectives of the study, is automatically generated along with the required local demand, the traversal OD matrix if necessary.
Figure 4: An example of multilevel network representation: node-link (left), detailed geometry (right)

Figure 5: An example of multilevel network representation: traffic assignment (left), mesoscopic simulation (centre), microscopic simulation (right)
3.1 Definition of the network for the subarea and its traversal

The window containing the subnetwork object of study is defined by the analyst using a graphic editor of polygons, the basic input defining graphically a sub-network is a set of closed polygonal lines that define a connected area. Just identifying which nodes fall inside the Problem Network can then derive the elements making up the sub-network. The rules for the automatic calculation of the traversal OD matrix, illustrated in Figure 8, are the following:

The sections of the sub-network must be the sections of the global area that have their starting node and/or their ending node with coordinates lying in the interior of the region that defines the Problem Network. All the attributes of the sections in the sub-network are inherited from the corresponding ones in the global area. The nodes of the sub-network model are those with coordinates lying in the Problem Network. The sections of the sub-network that have starting node with coordinates in the Problem Network and the ending node outside it define an exit from the sub-network. The sections of the sub-network that have ending node with coordinates in the Problem Network and starting node outside it define an entry to the sub-network.
The centroids of the sub-network are defined by the sections of the global network model with either the starting node or the ending node inside the Problem Area, and also by centroids of the global network with at least a connector attached to a node with coordinates lying in the Problem Network. If all the connectors of the centroid have attachment points lying in the Problem Network then, the centroid must be considered as a **centroid fully interior** to the Problem Network. In this case, a new centroid is generated for each of the attachments (see, in Figure 7, centroid 23, which corresponds to centroids 23a, b and c). Otherwise, i.e. if there are connectors with attachments not in the Problem Network, then new centroids are defined only for the attachments to sections or nodes inside the Problem Network (see, in Figure 8 centroid number 21, which corresponds to gate number 17 and 18 in the sub-network model).

There cannot be duplicates in numbering the gates of a sub-network model. Gates are attached to the corresponding starting or ending nodes of physical sections by means of connectors.

### 3.2 Estimation of the traversal OD matrix for the selected scenario

The local OD matrix for the scenario for that period of time contains the number of trips $t_{ij}$ between each origin $i$ and each destination $j$ for each time period. There will be two types of origins and destinations: the ones that lay in the borders of the area spanned by the network, corresponding to the input and output gates defined by the border of the sub-network, and the ones located inside the area defined by the sub-network. This is the situation schematized in Figure 8 explained below.

Given an O/D matrix for the whole area and a sub-network, the procedure to calculate the traversal starts by calculating the traversal O/D flows between gates defined by the border of the sub-network, that is, it extracts from the global O/D matrix the sub-matrix corresponding to the selected sub-network. This sub-network defines the scenario selected by the operator, where the traffic conflicts have been identified. The so-called traversal matrix is the local O/D matrix for the shaded area inside the rectangle,
corresponding to a sub-network of the road network for the whole area. The traversal matrix is composed by the original Origins and Destinations in the area plus some extra origins and destinations generated from the input and output gates of the flows into, from and through the area. In Figure 9, I/O,
 and I/O,
 correspond to the i-th and j-th input/output gates, which then generate the new centroids, corresponding to the flows form centroid r to centroid s crossing the area. I,
 is the k-th input gate for the flows with origin at centroid p, outside the area, that finish the trip inside the area, and O,
 the n-th output gate, for flows generated at a centroid inside the area that leaving the area through this output gate and finish the trip in centroid q outside the area.

Figure 8: Windowing into a sub area: sub network identification and traversal calculation

Figure 9: Traversal O/D matrix for a sub-area
The generation of traversal matrices is a standard procedure in AIMSUN NG supported efficiently by the described software architecture. After defining the selected subarea graphically (red dotted window), which can have any shape as the figure illustrates, the Traversal Generation algorithm is invoked, as depicted in Figure 10, activating the corresponding dialogue. In Figure 11 the centroids automatically generated by the procedure of the sub-network definition are also depicted.

The procedure starts by establishing the correspondence between gates and zones. The links considered as **in-gates** are all the outgoing connectors from the centroids located in the selected scenario, as well as all the links that enter the scenario boundaries. The links considered as **out-gates** are all the incoming connectors to the centroids located in the scenario, as well as all the links that exit the scenario boundaries. All the streets that cross the scenario boundary are assigned centroid numbers and are defined as directional gates. The utilities implemented in the AIMSUN NG Planner perform all these functions automatically for the Problem Network under study once it has been graphically defined as described.

### 3.3 Hybrid alternatives meso-micro

The process defined so far can be applied to the macro-meso or meso-micro combinations. The calculation of the traversal, and its subsequent adjustment from flow counts if available in the subarea, Codina and Barceló 2004, for the fine tuning of the local OD matrix, namely when time variability of the demand is taken into account, Barceló et al. 2004a, can also be done with the available tools in the integrated environment. This process makes sense when the studied subarea is isolated. However, a different situation arises when the analysis is done in the framework of traffic management, Barceló et al.
2004b, in this case the interest of the analysis is in the study of the subarea related to the rest of the network, for instance to account for diversions of traffic flows around the subarea consequence of congestion or incidents in the subarea.

In this case at a given time during the simulation it may happen that a fraction of the trips with origins and destinations outside the subarea that initially were traveling in paths across the subarea will be diverted to paths bypassing the subarea as illustrated in Figure 12. A way of dealing with this situation is simulating mesoscopically the whole network and microscopically the subarea network.

This raises additional consistency problems when the Mesoscopic simulator and the microscopic simulator are two independent pieces of software. The integration of Meso and Micro simulator has been analyzed in detail by Burghout 2004, and Burghout et al. 2005, in the case of MIME and MITSIM. In particular the integration has to solve the consistency problems in route choice and network representation, traffic dynamics at meso-micro boundaries, traffic performance for micro and meso models and communication and data exchanges. The network representation consistency problems has already been discussed before, but the other problems raise basically from the fact that vehicles in the meso model and vehicles in the micro model are not the same, and the path calculation is different.

However, software architecture as the one presented in Section 2 overcomes all these drawbacks given that network representations share the same object model and model database and vehicles can be unique and the same for meso and micro if meso model is based on an approach that individualized
vehicles, furthermore path calculation can be carried out by a common “path calculator server” and are the same at both levels.

![Diagram of Upper and Lower Level Macro Models](image)

Figure 12: Path diversion from Origin r to Destination s bypassing the subarea (dotted line)

4. A COMPUTATIONAL FRAMEWORK FOR TRAFFIC EQUILIBRIUM MODELS

4.1 Static User Equilibrium

According to Florian and Hearn 1995, “traffic equilibrium models are descriptive models with the aim to predict the link flows and travel times that result from the way in which travelers choose routes from their origins to their destinations on a transportation network”. The equilibrium model consists essentially of:

- A model of the transportation infrastructure in terms of a graph $G=(N,A)$, whose nodes $i \in N$ represent either “centroids”, that is the sources or origins of trips and the sinks or destinations of these trips, or intersections. The links $a \in A$ represent the physical infrastructure, that is the road or urban street sections between intersections, or they are “dummy links” also called “connectors” whose role is to connect physically the origin and destination nodes to the graph modeling the road network. The graph can be interpreted as an abstraction of the physical road network.
- Link cost functions that model the time delay for travel on an arc, which can be expressed in terms of volumes in the arc, generalized transportation costs or in terms of the concept of cost relevant for the intended analysis.
- The specification of the demand in terms of a trip or Origin-destination matrix.
- A behavioral modeling hypothesis that translates the assumptions on how traveler choose their routes from origins to destinations in the network. User Equilibrium models are based on Wardrop’s 1952, 1st Principle, which states that "For each O-D pair, at user equilibrium, the travel time on all used paths is equal, and (also) less than or equal to the travel time that would be experienced by a single user on any unused path”

Traffic assignment is the process which allocates the trips to the transportation network according to the modeling hypothesis, The network user equilibrium model, based on first Wardrop’s principle, is formulated, in the space of the path flows, by supposing that for every O/D pair Wardrop’s user optimal principle is satisfied, or in other words, that all the used directed paths are of equal cost, that is:

$$s_k - u_i = \begin{cases} 0 & \text{if } h_k \geq 0 \\ \geq 0 & \text{if } h_k = 0 \end{cases} \quad k \in K; i \in I$$

(1)
where \( \mathfrak{I} \) is the set of all origin/destination pairs, O/D, \( K_i \) is the set of paths for O/D pair \( i \), \( s_k \) is the cost of path \( k \), \( u_i \) is the cost of the least cost path for any O/D pairs \( i \):

\[
\bar{u}_i = \min \{ s_k \mid k \in K_i \} \quad \forall i \in \mathfrak{I}
\]

And the flows on paths \( k \), \( h_k \), satisfy flow conservation and non-negativity conditions:

\[
\Omega = \left\{ h_k \mid \sum_{k \in K_i} h_k = g_i, h_k \geq 0, k \in K_i, \forall i \in \mathfrak{I} \right\}
\]

(2)

Where \( g_i \) is the demand for the \( i \)-th OD pair. The network equilibrium model may be stated in the form of a variational inequality, Florian and Hearn, 1995:

\[
\left( s^*_k - u^*_i \right) \left( h_k - h^*_k \right) \geq 0, k \in K_i, i \in \mathfrak{I}
\]

(3)

where \( h^*_k \) is any feasible path flow. If \( h^*_k > 0 \), then \( s^*_k = u^*_i \) since \( h_k \) may be smaller than \( h^*_k \), if \( h^*_k = 0 \) then the inequality is satisfied when \( s^*_k - u^*_i \geq 0 \). By summing over \( k \in K_i \), and \( i \in \mathfrak{I} \), and taking into account constraints (2) when the demand \( g_i \) is constant, model (3) can be reformulated as follows (Fisk and Boyce, 1983) (Magnanti, 1984) (Dafermos, 1980):

\[
\left( v^* - v \right) ^T \left( \delta - v \right) \geq 0
\]

(4)

Which is the variational inequality formulation derived by Smith (Smith, 1979).

When the user cost functions are separable, that is, they depend only on the flow in the link: \( s_a(v) = s_a(v_a) \) \( a \in A \), and demands \( g_i \) are considered constant, independent of travel costs, the variational inequality formulation has the following equivalent convex optimization problem (Patriksson, 1994; Florian and Hearn, 1995):

\[
\min S(v) = \sum_{a \in A} \int_0^{v_a} s_a(x) \, dx
\]

s.t. \( \sum_{k \in K_i} h_k = g_i, \forall i \in \mathfrak{I} \)

\[
\sum_{k \in K_i} h_k \geq 0, k \in K_i, i \in \mathfrak{I}
\]

(5)

and the definitional constraint of \( v_a = \sum_{i \in \mathfrak{I}} \sum_{k \in K_i} \delta_{ak} h_k, \forall a \in A \) where \( \delta_{ak} = \begin{cases} 1 & \text{if link} \ a \ \text{belongs to path} \ k \\ 0 & \text{otherwise} \end{cases} \)

This is the version implemented in AIMSUN NG, in the “Transport Planning” component of Figure 1. The optimization problem is solved by an improved version of the linear approximation method of Frank and Wolfe, (Patriksson, 1994; Florian and Hearn, 1995).

4.2 Dynamic User Equilibrium

From an analytical point of view dynamic traffic assignment has been usually related to the concept of the dynamic user equilibrium problems. Some of the most successful approaches are inspired on the seminal paper by Friesz et al. 1993 that proposes a dynamic network user equilibrium model which equilibrates the disutility of the temporal choices. To achieve such equilibrium they take the perspective “that the essential choices available to users of a transportation network —route choice and departure time — occur in time-varying environments and are made rationally”, and they conclude with the assumption that these rational choices can only be made if the disutility of the alternatives are equilibrated.

Two main approaches have been used to model these route choices. One based on a generalization of Wardrop’s first principle of static traffic assignment, in which users try to optimize their route based on the current information, this approach describes the evolution of flows when users make route choice decisions based on experienced travel times, and it is usually known as a preventive or en-route
assignment, it does not achieve a day-to-day equilibrium pattern, therefore it is considered a dynamic traffic assignment principle and not a true equilibrium. In the above referenced paper Friesz et al. 1993, propose an alternative generalization of Wardrop’s principle stated in the following terms: If, at each instant in time, for each OD pair, the flow unit costs on utilized paths are identical and equal to the minimum instantaneous unit path cost, the corresponding flow pattern is said to be in dynamic traffic equilibrium. This approach, also known as reactive assignment, can be interpreted in terms that could correspond to users having access to a real-time driver information traffic forecasting system or, alternatively, as an approximation to a process by which traveler combine the experienced travel times with conjectures to forecast the temporal variations in flows and travel costs.

In the above reference the dynamic equilibrium problem is formulated in the space of path flows \( h_k(t) \), for all paths \( k \in K_i \), the set of feasible paths for the \( i \)-th OD pair at time \( t \). The path flow rates in the feasible region \( \Omega \) satisfy at any time \( t \in (0,T) \) the flow conservation and non-negativity constraints:

\[
\Omega = \left\{ h(t) \mid \sum_{k \in K_i} h_k(t) = g_i(t), i \in I; h_k(t) \geq 0 \right\}
\]

for almost all \( t \in (0,T) \)

where \( I \) is the set of all OD pairs in the network, \( T \) is the time horizon, and \( g_i(t) \) is the fraction of the demand for the \( i \)-th OD pair during the time interval \( t \). The approaches assume that the optimal user equilibrium conditions can be defined as, Friesz et al. 1993, a temporal version of the static Wardrop user optimal equilibrium conditions, which can be formulated as:

\[
s_k(t) = \begin{cases} 
  u_i(t) & \text{if } h_k(t) > 0 \\
  u_i(t) & \text{Otherwise}
\end{cases} \quad \text{for } \forall k \in K_i, \forall i \in I, \text{for almost all } t \in (0,t) \quad h_k(t) \in \Omega
\]

\[
u(t) = \min_{k \in K_i} s_k(t)
\]

Where \( s_k(t) \) is the path travel time on path \( k \) determined by the dynamic network loading. Friesz et al. 1993, show that these conditions are equivalent to the variational inequality problem consisting on finding \( h \in \Omega \) such that:

\[
[S(h)]h - h^* \geq 0, \forall h \in \Omega
\]

According to (Florian et al., 2001), a dynamic traffic assignment model consists of two main components:

1. A method to determining the path dependent flow rates on the paths on the network, and
2. A Dynamic Network Loading method, which determines how these path flows give raise to time-dependent arc volumes, arc travel times and path travel times

In the Dynamic Network Loading, also known as Dynamic Network Flow Propagation, (Cascetta, 2001), “models simulate how the time-varying continuous path flows propagate through the network inducing time-varying in-flows, out-flows and link occupancies”. A wide variety of approaches, from analytical (Wu, 1991; Wu et al., 1998a; Wu et al., 1998b; Xu et al., 1998; Xu et al., 1999), to simulation based, (Florian et al., 2001), have been proposed. In all them path flows are determined by an approximate solution to the mathematical model for the dynamic equilibrium conditions. The differences between the various referenced approaches lay in the discretization scheme used to solve (3), and the algorithmic approach to solve the discretized problem (i.e. projection algorithms, successive averages, etc.) and the dynamic network loading mechanism, analytical (Wu, 1991; Wu et al., 1998a; Wu et al., 1998b; Xu et al., 1998; Xu et al., 1999), or simulation based (Florian et al. 2001).

This computational framework, whose diagram is depicted in Figure 13, can be implemented in practice in various ways, depending on whether the path selection is based on a direct numerical solution of (8) or
on a stochastic route choice model, and on whether the dynamic network loading is implemented analytically or based on a simulation mechanism. We propose two heuristic approaches:

1. Implement the route choice on basis to the dynamic user equilibrium conditions (7) solving (8) numerically, and perform an approximate network loading mechanism based on a mesoscopic traffic simulation, and

2. Implement the route choice on basis to a discrete stochastic route choice model, and base the Dynamic Network Loading mechanism on microscopic simulation with AIMSUN, Barceló and Casas 2004.

Figure 13 Conceptual framework for the computational implementation of Dynamic Traffic Assignment

5. HEURISTIC DYNAMIC ASSIGNMENT BASED ON MESOSCOPIC TRAFFIC SIMULATION

This corresponds to the first alternative described above, implementing the path choice as defined in (7) by the dynamic version of Wardrop’s user equilibrium principle, solving the variational inequalities (8), and performing the Dynamic Network Loading by means of a mesoscopic traffic simulation. Computationally this means to apply an iterative similar to those proposed by Fukushima 1992, or Wu 1991, Wu et al. 1998a and 1998b, although inspired on the computational results of Florian et al. 2002, and Mahut et al. 2003, 2004, we have also adopted the Method of Successive Average (MSA), instead of the projection method to solve the variational inequalities problem in (8).
To solve numerically the problem the simulation horizon $T$ has been discretized into discrete time periods $t = 1, 2, \ldots, \frac{T}{\Delta t}$ of length $\Delta t$, corresponding to equilibrium flows according to (6) and (7) where the feasible flows $h_k(t)$ are the solution of the discretization of (8):

$$\sum_{t} \sum_{k} s_k(t)[h_k(t) - h_k^n(t)] \geq 0$$

(9)

Where $K = \bigcup_{i \in I} K_i$ is the set of all paths for all OD pairs. The MSA algorithm works as follows:

**Step 0:** Initialization: set the iterations counter $n:=0$
- Compute a set of the K shortest paths (see section 6.2) for each time interval $t$ based on the Initial Costs, and load the demands $g_i(t)$ to obtain an initial feasible solution.

**Step 1:** Update the iterations counter, set $n:=n+1$
- If $n<N$: add new shortest paths: calculate a new shortest path for each time interval $t$, for each OD pair $i \in I$, and assign to each path $k \in K_i$ the input flow $g_i(t)$
- Else: identify the shortest among the used paths for each OD pair $i \in I$, (at most $N$ alternative paths for each OD pair will be available, and no new paths are calculated) let $u_i^n(t)$ the cost of the shortest path for OD pair $i$, then redistribute the flows according to the MSA:

$$h_k^*(t) = \begin{cases} h_k^{n-1}(t) \left(\frac{n-1}{n}\right) + \frac{g_i(t)}{n} & \text{if } s_k^{n-1}(t) = u_i^{n-1}(t) \\ h_k^{n-1}(t) \left(\frac{n-1}{n}\right) & \text{otherwise} \end{cases}$$

(10)

For $k \in K_i$, $i \in I$ and all $t$

**Step 2:** Dynamic Network Loading: load the demands onto the network executing the mesoscopic traffic simulation, to obtain the new link costs.

**Step 3:** Convergence test based on the relative gap function:
- If $\text{Rgap}^n(t) \leq \varepsilon$ then stop
- Else, return to Step 1

A way of measuring the progress towards the equilibrium in an assignment, and therefore qualify the solution, is the relative gap function, $\text{Rgap}(t)$, Florian et al. 2001, Janson, 1991, that estimates at time $t$ the relative difference between the total travel time actually experienced and the total travel time that would have been experienced if all vehicles had the travel time equal to the current shortest path:

$$\text{Rgap}(t) = \frac{\sum_{i \in I} \sum_{k \in K_i} h_k(t)[s_k(t) - u_i(t)]}{\sum_{i \in I} g_i(t)u_i(t)}$$

(11)

Where $u_i(t)$ are the travel times on the shortest paths for the $i$-th OD pair at time interval $t$, $s_k(t)$ is the travel time on path $k$ connecting the $i$-th OD pair at time interval $t$, $h_k(t)$ is the flow on path $k$ at time $t$, $g_i(t)$ is the demand for the $i$-th OD pair at time interval $t$, $K_i$ is the set of paths for the $i$-th OD pair, and $I$ is the set of all OD pairs.

The mesoscopic simulation model has been based on the event scheduling approach, as in Burghouwt 2004, and Mahut et al. 2003. This has proven to be a computationally efficient solution when the simulation is not based on tracking individually the vehicles all the time, but in describing approximately their trajectories in the links, this means that only the generation of new vehicles, the entrance of a
vehicle into a link, or the transfer of a vehicle from one link to the next according to the turning
movements at intersections, are the events of interest that must be taken into account.

Once this approach has been decided the next step is to decide the approach to simulate the flow
dynamics. Basically there are two main approaches to mesoscopic traffic simulation: those in which
individual vehicles are not taken into account, and vehicles are packed in packages or platoons (although
platoons may have size one) that move along the links, as in CONTRAM, Leonard et al 1989; and those
in which flow dynamics is determined by simplified dynamics of individual vehicles, as DYNASMART,
account that one of our primary goals was to embed the mesoscopic simulator in the integrated
environment of AIMSUN NG described in section 2, and integrate it with the microscopic simulator and
even more, make possible the hybrid meso-micro simulation as described in section 3.3 it was natural to
adopt the vehicle based approach in our developments.

A summary description of the mesoscopic approach follows. The approach models the link as in MEZZO,
Burghout 2004, and other simulators, splitting the link in two parts, the running part and the queue part,
Figure 14. The running part is that part of the link where vehicles are not yet delayed by the queue
spillback at the downstream node, where the capacity is limited by stop, give way or traffic lights.

![Figure 14 Link model](Link (i,j) Running Part \ Link (i,j) Queue Part)

Nodes are modeled according to a queue server approach, to account explicitly with traffic lights and the
delays that they cause, and the interactions between traffic flows at intersections. Individual vehicle
dynamics in the running part is approximated by a simplified car following model compatible with the
macroscopic speed-density relationship on the link. This speed is used to estimate the earliest time at
which the vehicle could exit the link, unless it is affected by the queue spill back when reaching the border
between the running part and the queue part, the vehicle dynamics is then ruled by the queue discharging
process, that determines the time at which the event “exit the link” (and “enter the next link”) will occur.
The boundary between the running part and the queue part is dynamic according to the queue spillback
and queue discharge processes.

A similar approach had been adopted in an earlier version of a mesoscopic simulator, PACKSIM,
developed by the authors, Barceló and Grau 1991, 1994. However, this approach, designed to servicing
an adaptive traffic control strategy based on a rolling horizon method, although sufficient for the short
term simulation for the control purposes, was inadequate for a straightforward extrapolation to general
simulation purposes. PACKSIM, although distinguishing between a running or free part of the link, and a
queue part, was based on a packet dynamics, although packet sizes could be of one vehicle, and
therefore adapted to an individual vehicles approach, and vehicles had not origins and destinations, and
therefore were not traveling on routes across the network. Nevertheless these two drawbacks could be
easily overcome; the major obstacles were that the seminal ideas on a simplified car-following and queue
server to deal with the dynamics of the running and queue parts respectively required a substantial
improvement for the new purposes.

The starting point was the safe to stop approach to car-following models to describe the dynamic of
leader-follower couples, Gerlough and Huber 1975, (see also Mahut 1999a and 1999b) according to this
approach, the safe distance between the leader vehicle \( n \) and the follower vehicle \( n+1 \) at time \( t \) can be modeled as:

\[
s(t) = x_n(t) - x_{n+1}(t) = d_1 + d_2 + L - d_3
\]  

(12)

where \( x_n(t) \) and \( x_{n+1}(t) \) are the positions of vehicles \( n \) and \( n+1 \) respectively at time \( t \), \( d_1 \) is the distance traveled by the follower vehicle \( n+1 \) during the reaction time \( T \) at speed \( u_{n+1}(t) \) (\( d_1 = T u_{n+1} \)); \( d_2 \) and \( d_3 \) are respectively the distances traveled by the leader vehicle \( n \) and the follower vehicle \( n+1 \) during the deceleration maneuver, and \( L \) is the effective vehicle length (vehicle length plus the minimum inter-vehicle distance). Assuming steady state conditions, if vehicles in the running part of the link are traveling at the same average speed, and one accepts the simplifying hypothesis that vehicles have similar breaking capabilities, then \( d_2 = d_3 \), and the minimum safe spacing between vehicles is:

\[
s(t) = x_n(t) - x_{n+1}(t) = d_1 + L = T u_{n+1} + L
\]  

(13)

It can be easily proven that with this simplifying hypothesis the basic car-following model reaction equals a sensitivity times the stimuli, depending on the hypothesis on the sensitivity coefficient lead to the well known speed-density relationships of the Greenshields and Greenberg type:

\[
u(k) = u_f \left( 1 - \frac{k}{k_j} \right)
\]

\[
u(k) = u_c \ln \left( \frac{k_j}{k} \right)
\]  

(14)

Where \( u_f \) and \( u_c \) are the free flow speed and speed at capacity respectively, and \( k_j \) is the jam density. This model can be complemented after the empirical evidence that there are two limiting densities \( K_{\text{min}} \) and \( K_{\text{max}} \) which represent respectively the minimum and maximum densities where the speed is still a function of the density, Del Castillo and Benitez, 1995. Similar considerations have been used in the above mentioned mesoscopic simulators MEZZO and DYNAMIT. This simplified model, consistent with the basic assumptions of traffic flow theory can be used to model flow dynamics at the running part of the link and schedule the events arrival and exit to a link.

With respect to the queue part we assume that the boundary moves at the queue clearance wave speed according with Akçelik's model, Akçelik 1999:

\[
v_x = \frac{v_n q_{\text{max}} L}{1000 v_n - q_{\text{max}} L}
\]  

(15)

Where \( v_n \) the maximum queue discharge speed, and \( q_{\text{max}} \) maximum flow discharge are intersection dependent parameters that are a function of the signal timings in the case of signalized intersections. Akçelik's model's also estimate the queue discharge speed, that will be the speed of vehicles in the queue and therefore will determine the link exit time depending on their position in the queue as:

\[
v_p(t) = v_n \left[ 1 - e^{-m_p (t-t_i)} \right]
\]  

(16)

Where \( v_p(t) \) is the queue discharge speed at time \( t \) since the start of the displayed green period for the corresponding phase at the traffic light, \( t_i \) is the vehicle start response time (or reaction time at stop), and \( m_p \) is an intersection dependent parameter. This model have been empirically validated with AIMSUN microscopic simulator in the benchmark proposed by Akçelik at the Highway capacity Committee Half year meeting in Truckee in 2001, Akçelik 2001.

Vehicles travel from origins to destinations across the network along time dependent paths that change with time according to changes in traffic conditions, this means that vehicles entering a link must go into the suitable lane to perform at the end node the turnings required by vehicle's path. Vehicles are
positioned in the correct lane at the beginning of the link and a variable penalty factor is added in the computation of the time to exit the link to emulate the lane changes or merging that the vehicle should have been done to position itself in the adequate position.

A preliminary version, currently in the beta testing process, has been implemented in AIMSUN NG as one of the available traffic analysis tools, as illustrated in Figure 2. Figure 15 depicts the dialogue to activate the mesoscopic simulation of the selected network model.

Figure 15: Activating a mesoscopic simulation in AIMSUN NG

The preliminary prototype of mesoscopic simulator is being currently tested against AIMSUN microscopic simulator in order to check not only the quality of the results but also the consistency between both simulators. Figure 16, compares the micro and meso results in terms of flows during the simulation for a short link of a roundabout, meso showing a smoother behavior.
Figure 16: Comparing Meso and Micro Simulation: behavior at a roundabout with short links

Figure 17: Mesoscopic link average densities on the left, microscopic on the right
Figure 17 provides a qualitative comparison between traffic densities on the links, showing an acceptable degree of similarity between the meso and micro simulations. Figure 18 depicts the quantitative comparison of meso and micro flows for a long section, showing that in this case the level of agreement is very high, qualitative appraisal that is confirmed by the quantitative analysis in Table 1.

The main discrepancies appear in the first half hour. Table 1 summarizes the main statistics of the time series of Figure 17. The analysis is based on Theil’s inequality coefficients to measure the degree of similarity of two time series, Barceló and Casas 2004, Theil 1966.
<table>
<thead>
<tr>
<th>Mesoscopic Flow</th>
<th>Microscopic Flow</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum</td>
<td>1200</td>
</tr>
<tr>
<td>Maximum</td>
<td>1932</td>
</tr>
<tr>
<td>Mean</td>
<td>1671</td>
</tr>
<tr>
<td>Standard Deviation</td>
<td>182.43</td>
</tr>
<tr>
<td>U-Theil</td>
<td>0.065</td>
</tr>
<tr>
<td>Um-Theil</td>
<td>0.371</td>
</tr>
<tr>
<td>Us-Theil</td>
<td>0.147</td>
</tr>
<tr>
<td>Uc-Theil</td>
<td>0.54</td>
</tr>
</tbody>
</table>

Table 1: Comparison between meso and micro flows for a long section.

6. HEURISTIC STOCHASTIC DYNAMIC TRAFFIC ASSIGNMENT BASED ON MICROSCOPIC SIMULATION

The simulation approach used is a route based microscopic simulation in which, vehicles are input into the network according to the demand data defined as an O/D matrix (preferably time dependent) and travel across the network following specific paths in order to reach their destination. In the route based simulation new routes are to be calculated periodically during the simulation, and a Route Choice model is needed, when alternative routes are available to determine how the trips are assigned to these routes.

The key question that this approach raises is whether this simulation can be interpreted in term of a stochastic heuristic dynamic traffic assignment or not. We propose to investigate the answer to this question in the case of a microscopic simulation using AIMSUN 2005, a route based microscopic simulator (Barceló et al. 1995, 1998). This paper is more elaborated version of a previous research reported in Barceló and Casas 2002 and 2004, and Barceló 2004. This process can be interpreted in terms of a heuristic approach to dynamic traffic assignment similar to the one proposed by Florian et al. 2001, consisting on:

1. A method to determining the path dependent flow rates on the paths on the network, based on a Route Choice function, and
2. A Dynamic Network Loading method, which determines how these path flows give raise to time-dependent arc volumes, arc travel times and path travel times, heuristically implemented by microscopic simulation.

The implemented simulation process, Barceló and Casas 2004, and Barceló et al. 1995, based on time dependent routes consists of the following procedure, whose conceptual diagram is depicted in Figure 5, is as follows:

The following algorithm describes how the stochastic heuristic dynamic traffic assignment is implemented:

**Step a: n = 0**

**Step b: Simulation Model**

**Step 0.** Calculate initial shortest path(s) for each OD pair using the defined initial costs

**Step 0.1. Initialization:**

Evaluate Initial Cost Function per each link $j$:

for each $j \in 1...L$: $Cost_j = InitialCost_j$

**Step 0.2. Apply Shortest Path routine:**

for each destination centroid $d$:

Calculate Shortest Path Tree $SPT_d$ using $Cost_j, j \in 1...L$

**Step 0.3. Identify Shortest Path from Shortest Path Tree:**

for each OD pair $i$ (from origin centroid $o$ to destination $d$)

Add to Path(s) $SP_{con}$ to $K_i$

**Step 0.4. Read Expected and Experimented Link Cost of Replication n-1**

if $n = 0$ then Expected and Experimented Link Cost is the travel time in free flow conditions otherwise is read from Replication n-1
Step 1: Network Loading
Simulate for a predefined time interval $\Delta t$ assigning to the available path $K_i$ the fraction of the trips between each OD pair $i$ for that time interval according to the route choice model.

Step 2: Shortest Path Calculation
Recalculate shortest path, taking into account the experimented average link travel times.

Step 2.1. Update Link Cost Functions:
Store Cost Link per each link $j$ as Experimented Cost at iteration $n$ at time interval $t-1$.
Evaluate dynamic cost function per each link $j$:
for each $j \in 1... L : Cost_j$ is a combination of the expected and experimented cost of link $j$
Store cost link per each link $j$ as expected cost at iteration $k$ at time interval $t$.

Step 2.2. Apply shortest path routine:
for each destination centroid $d$:
Calculate shortest path tree $SPT_d$ using $Cost_j$ $j \in 1... L$

Step 2.3. Identify shortest path from shortest path tree:
for each OD pair $i$ (from origin centroid $o$ to destination $d$)
Add to Path(s) $SP_{con}$ to $K_i$

Step 3. If there are guided vehicles or variable message signs that suggest rerouting, provide the drivers who are allowed to dynamically reroute during a trip with the information calculated in Step 2.

Step 4. If all the demand has been assigned, then stop. Otherwise, go to Step 1.

Step c. Update Iteration counter $n = n + 1$

Step d: If the Convergence Criterion is not fulfilled, go to Step b.

This algorithm implements a heuristic approach to equilibrium based on repeating the simulation scheme a number of times and defining a link cost function including predictive terms, as proposed by Friesz et al. 1993, Xu et al. 1998 and 1999, that is a reactive dynamic assignment, trying to achieve an equilibrium and not become merely a dynamic assignment procedure, therefore the costs used in the path calculation must include a “forecasting” component. To achieve this objective, the simulation is replicated $N$ times and link costs for each time interval and every replication are stored and thus at iteration $l$ of replication $j$ the costs for the remaining $l+1$, $l+2$, ..., $L$ (where $L=T/\Delta t$, being $T$ the simulation horizon and $\Delta t$ the user defined time interval to update paths and path flows) time intervals for the previous $j-1$ replications can be used in an anticipatory day-to-day learning mechanism to estimate the expected link cost at the current iteration. Let $s_{a}^{j,l+i}(v)$ be the current cost of link $a$ at iteration $l$ of replication $j$, then the average link costs for the future $L-l$ time intervals, based on the experienced link costs for the previous $j-1$ replications is:

$$s_{a}^{j,l+i}(v) = \frac{1}{f-1} \sum_{m=1}^{f-1} s_{a}^{m,l+i}(v), \quad i = 1,\ldots,L-l, (17)$$

The “forecasted” link cost can then be computed as:

$$\overline{s}_{a}^{j,l+1}(v) = \sum_{i=0}^{L-l} \alpha_i s_{a}^{j,l+i}(v), \quad \text{where} \quad \sum_{i=0}^{L-l} \alpha_i = 1, \alpha_i \geq 0, \forall i; \text{ are weighting factors} \quad (18)$$

The resulting cost of path $k$ for the $i$-th OD pair is

$$\overline{s}_{k}^{l+1}(v) = \sum_{a \in A} \overline{s}_{a}^{j,l+1}(v) e_{ak} \quad (19)$$
where, as usually $\delta_{ak}$ is 1 if link a belongs to path k and 0 otherwise. The path costs $\bar{S}_{k}\left(h^{l+1}\right)$ are the arguments of the route choice function (logit, C-logit, proportional, user defined, etc.) used at iteration $l+1$ to split the demand $g_i^{l+1}$ among the available paths for OD pair i. In the computational experiments discussed in this paper a simplified version consisting of a link cost function defined as:

$$c_i^{l+1} = \lambda c_i^k + (1-\lambda)\bar{c}_i^k$$

(20)

Where $c_i^{k+1}$ is the cost of using link i at time t at iteration $k+1$, and $c_i^k$ and $\bar{c}_i^k$ correspond respectively to the expected and experienced link costs at this time interval from previous iterations. Figure 19 illustrates how the link cost function is evaluated for each iteration and each time interval.

Figure 19: Scheme of link cost function evaluation

6.1 Discrete Route Choice Models

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

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

The simulation experiments reported in this paper have been implemented in AIMSUN selecting the Logit, C-Logit and Proportional route choice functions from the default route choice functions available in the simulator. The Multinomial Logit route choice model defines the choice probability $P_k$ of alternative path $k, k \in K_i$, as a function of the difference between the measured utilities of that path and all other alternative paths:

$$P_k = \frac{e^{\theta V_k}}{\sum_{l \in K_i} e^{\theta V_l}} = \frac{1}{1 + \sum_{l \in K_i} e^{\theta (V_l - V_k)}} \quad (21)$$

where $V_i$ is the perceived utility for alternative path $i$ (i.e. the opposite of the path cost, or path travel time), and $\theta$ is a scale factor that plays a two-fold role, making the decision based on differences between utilities independent of measurement units, and influencing the standard error of the distribution of expected utilities, determining in that way a trend towards utilizing many alternative routes or concentrate in very few routes, becoming in that way the critical parameter to calibrate how the logit route choice model leads to a meaningful selection of routes or not.

A drawback reported in using the Logit function is the observed tendency towards route oscillations in the routes used, with the corresponding instability creating a kind of flip-flop process. According to our experience there are two main reasons for this behavior. The properties of the Logit function, which and the inability of the Logit function to distinguish between two alternative routes when there is a high degree of overlapping.

The instability of the routes used can be substantially improved when the network topology allows for alternative routes with little or no overlapping at all, playing with the shape factor of the Logit function and re-computing the routes very frequently. However, in large networks where many alternative routes between origin and destinations exist, and some of them exhibit a certain degree of overlapping the use of the Logit function may still exhibit some weaknesses. To avoid this drawback the C-Logit model, (Cascetta et al., 1996; Ben-Akiva and Bierlaire, 1999), has been implemented. In this model, the choice probability $P_k$, of each alternative path $k$ belonging to the set $K_i$ of available paths connecting the i-th OD pair, is defined by:

$$P_k = \frac{e^{\theta (V_k - CF_k)}}{\sum_{l \in K_i} e^{\theta (V_l - CF_l)}} \quad (22)$$

where $V_i$ is the perceived utility for alternative path $i$, i.e. the opposite of the path cost, and $\theta$ is the scale factor, as in the case of the Logit model. The term $CF_k$, denoted as ‘commonality factor’ of path $k$, is directly proportional to the degree of overlapping of path $k$ with other alternative paths. Thus, highly overlapped paths have a larger CF factor and therefore smaller utility with respect to similar paths. $CF_k$ is calculated as follows:

$$CF_k = \beta \cdot \ln \sum_{rs \in L_k} \left( \frac{L_{1r} L_{2s} L_{1k} L_{2k}}{L_{1r} L_{1k} + L_{2s} L_{2k}} \right)^{\gamma} \quad (23)$$

where $L_k$ is the length of arcs common to paths $i$ and $k$, while $L_i$ and $L_k$ are the length of paths $i$ and $k$ respectively. Depending on the two factor parameters $\beta$ and $\gamma$, a greater or lesser weighting is given to the ‘commonality factor’. Larger values of $\beta$ means that the overlapping factor has greater importance with respect to the utility $V_i$; $\gamma$ is a positive parameter, whose influence is smaller than $\beta$ and which has the
opposite effect. The utility \( V_i \) used in this model for path \( i \) is the opposite of the path travel time \( t_{ti} \), (or path cost depending on how has been defined by the user).

Other option is the estimation of the choice probability \( P_k \) of path \( k, k \in K_i \) in terms of a generalization of Kirchoff's laws given by the function

\[
P_k = \frac{CP_k^{-\alpha}}{\sum_{l \in K_i} CP_l^{-\alpha}}
\]

where \( CP_i \) is the cost of path \( i \), \( \alpha \) is in this case the parameter whose value has to be calibrated.

### 6.2 Initial K-Shortest Paths

At the beginning of the simulation, using the Initial Cost function, one shortest path tree is calculated per each destination centroid, so during the first interval all vehicles are assigned to the same alternative. In order to start considering more than one alternative, as a way to anticipate the assignment process, at the beginning of the simulation, \( k \)-shortest path trees are calculated. The problem of enumerating, in order of increasing length, the \( k \) shortest paths has received considerable attention in the literature, Bellman 1960, Dreyfus 1969, Fox 1970, Eppstein 1994, 1999. The different algorithms to solve this problem Eppstein 1994,1999, Jimenez et al. 1999, Martins 1984, Martins et al. 1996, are based in, after computing the shortest path from every node in the graph, the algorithm builds a graph representing all possible deviations from the shortest path. Therefore, all \( k \) shortest path obtained as a result, use the same length arc, associated to a cost function, and in our case the calculation of the \( k \) shortest path has to anticipate the evolution of the arc costs considering the traffic flow assigned to each path.

![Figure 20: Generic scheme of K-Shortest Path Algorithm](image)

The algorithm calculates every iteration a new shortest path until the number of shortest path available reaches the parameter \( \text{MaxKSP} \). The figure 20 depicts the generic scheme of the algorithm that iteratively:

- evaluates the cost function in each arc (first iteration the cost function is the travel time in free-flow conditions)
- calculates a new shortest path
- determines the path flow (using an incremental loading procedure, described bellow) and update the flow in each arc

The components of this algorithm are:
- **Shortest Path Algorithm**
  The computation of the shortest path corresponds a variation of Dijkstra’s label setting algorithm.

- **Arc Cost Function**
  The cost of using arc a (sa(va)) is a function of the flow in arc a $v_a$, usually one of the BPR family as
  
  $$s_a(v_a) = t_a \left(1 + \beta \frac{v_a}{c_a} \right)^{\alpha}$$
  
  or similar ones used in the static user equilibrium assignment.

- **Incremental Loading Algorithm**
  The path flow rates in the feasible region $\Omega$ satisfy the conservation flow and non-negativity constraint (where the traffic demand of O-D pair $i$ is denoted by $g_i$ as before). That is:
  $$\Omega = h^i_k : \sum_{k \in K_i} h^i_k = g_i, i \in I; h^i_k \geq 0$$

  For each O-D pair $i \in I$ and path $h^i_k$, evaluate the path flow assigned to path $h^i_k$ $k \in K_i$ at iteration $n$ $h^i_k(n)$:
  $$\begin{align*}
  h^i_k(n) &= h^i_k(n-1) + \lambda (g_i - h^i_k(n-1)) \\
  h^i_l(n) &= h^i_l(n-1) - \lambda (h^i_l(n-1)), \quad l = 1 \ldots (k-1)
  \end{align*}$$

  where $\lambda = \frac{1}{n+1}$

  The algorithm to calculate the $k$-shortest path can be stated as follows ($n$ is the iteration index and $k$ is the shortest path index):

  **Step 0**: Initialization : $n=0$ and $k=1$
  
  Compute the $k$-th shortest path based on the free-flow travel times.
  
  For each O-D pair $i \in I$, assign $h^i_k(n) = g_i$

  **Step 1**: Compute the flow $v_a$ for each arc $a$:
  $$v_a = \sum_{i \in I} \sum_{k \in K_i} \delta^i_{a,k} h^i_k$$

  where $\delta^i_a = \begin{cases} 1, \text{arc } a \text{ belongs to path } k \text{ of O-D pair } i \\ 0, \text{otherwise} \end{cases}$

  Evaluate the cost function of each arc $a$ ($s_a(v_a)$).

  **Step 2**: $k= k + 1$, $n = n + 1$
  
  Compute the $k$-th shortest path based cost function of each arc $a$ ($s_a(v_a)$).

  Incremental Loading: For each O-D pair $i \in I$, evaluate:
  
  $$h^i_k(n) = h^i_k(n-1) + \lambda (g_i - h^i_k(n-1))$$

  for $l = 1$ to $(k-1)$
\[
    h_i^l(n) = h_i^l(n-1) - \lambda (h_i^l(n-1)), \quad l = 1 \ldots (k-1)
\]

where \( \lambda = \frac{1}{n+1} \)

**Step 3:** If \( k \) is equal to total number of shortest path \( \text{MaxKSP} \) then STOP
Otherwise, return Step 1

The figure 21 depicts an example of 3 initial \( k \)-shortest paths, in the model of the Preston network.

**6.3 Computational Results**

No formal converge proof can be given for the proposed heuristic stochastic dynamic assignment algorithm, since the heuristic network loading process based on microscopic simulation does not have an analytical form. A set of simulation experiments has been designed and conducted to explore empirically whether the described assignment process, depending on how it is implemented, can be associated to a heuristic realization of a preventive or a reactive dynamic assignment, assuming that a proper selection of a route choice model with the right values for the \( \theta, \beta, \gamma \) or \( \alpha \) parameters, depending on the model, should lead to the realization of some equilibrium. The progress towards equilibrium has been estimated as in the other cases, using the Rgap function (11).

The figures 22 and 23, and 24 and 25 respectively, depict the test networks, and the time evolution of the Rgap function for the logit route choice function, for the reactive version of the assignment procedure using the costs as defined in (20), using a K-shortest path algorithm, for the test models of:
- The borough of Amara in the City of san Sebastián in Spain. A model with 365 road sections, 100 nodes and 225 OD pairs.
- The model of Brunsviken network in Stockholm. This model has 493 road sections, 260 nodes and 576 OD pairs.

at iteration $k=20$, and $\lambda=0.25$, 0.5 and 0.75 respectively, for $\theta$ values of 30 in Amara, and 900 in Brunsviken. Rgap values tend almost to zero, as expected in equilibrium terms, and the variations for the various values of $\lambda$ show that $\lambda=0.75$ is the best.

Figure 22: AIMSUN model of Amara
Figure 23: Rgap for Amara (Reactive case)

Figure 24. AIMSUN model of Brunnsviken network in Stockholm
To determine the influence of the model parameters used to obtain these results, a set of complementary computational experiments was conducted. Table 1 depicts the summary of the most significant parameters for each route choice model and network model and the results of the method explored in this work enable guidelines to be established for the calibration process of each route choice model. In all experiments in which the shape factor (θ in the logit and the C-logit models and α in the proportional model) is a design factor, this factor becomes significant. Therefore, this parameter plays a relevant role during the calibration process. Focussing on each route choice model, the logit and C-logit function have more relevant parameters the aforementioned θ and the Initial K-SP, but in C-logit β must be added because it plays the same role as the scale factor.

To validate the results of the reactive version of the assignment procedure that uses the cost function $C_i^{k+1}(t) = \lambda C_i^k(t) + (1-\lambda) \hat{C}_i^k(t)$ (where $0 \leq \lambda \leq 1$ and $C_i^k(t)$ is the input cost of link i at iteration k at time interval t and $\hat{C}_i^k(t)$ is the output cost of link i at iteration k at time interval t, where the input cost $C_i^k(t)$ could be interpreted as the expected cost considered at interval t, while the output cost $\hat{C}_i^k(t)$ could be interpreted as the experimented or experienced link cost at the end of interval t), another set of computational experiments was conducted and a standard comparison between model and system outputs, using the GEH index and the validation based on the $RGap(t)$ function for Amara model were performed.

The GEH index for n pairs of (observed-simulated) values was calculated by the following algorithm:

For $i = 1$ to $n$ calculate

![Figure 25: Rgap for Brunnsviken (Reactive case)](image-url)
\[ GEH_i = \sqrt{\frac{2(ObsVal_i - SimVal_i)^2}{ObsVal_i + SimVal_i}} \]

If \( GEH_i \leq 5 \) Then \( GEH_i = 1 \)
Otherwise \( GEH_i = 0 \)
Endif
Endfor

\[ \text{Let } GEH = \frac{1}{n} \sum_{i=1}^{n} GEH_i \]

If \( GEH \geq 85\% \) then ACCEPT the model
otherwise REJECT the model
Endif

It needs to be noted that the GEH statistic is an “intuitive” and “empirical engineering” measure, not necessary a measure that a professional statistician would recognise or design to use. The criterion of 85% or 80% has been established by practitioners as a rule of thumb FHWA 2003, Traffic Appraisal Highways Agency 1966.

<table>
<thead>
<tr>
<th>AMARA</th>
<th>VITORIA</th>
<th>BRUNNSVIKEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOGIT</td>
<td>( \theta ) Initial K-SP</td>
<td>( \theta ) Initial K-SP</td>
</tr>
<tr>
<td></td>
<td>( \theta ) MaxNumberRoutes</td>
<td>( \theta ) MaxNumberRoutes</td>
</tr>
<tr>
<td></td>
<td>( \theta ) Initial K-SP * Initial K-SP</td>
<td>( \theta ) Initial K-SP * Initial K-SP</td>
</tr>
<tr>
<td></td>
<td>( \theta ) MaxNumberRoutes</td>
<td>( \theta ) MaxNumberRoutes</td>
</tr>
<tr>
<td>C-LOGIT with fixed ( \beta ) and ( \gamma )</td>
<td>( \theta ) Initial K-SP</td>
<td>( \theta ) Initial K-SP</td>
</tr>
<tr>
<td></td>
<td>( \beta ) Initial K-SP</td>
<td>( \beta ) Initial K-SP</td>
</tr>
<tr>
<td></td>
<td>( \beta ) * Initial K-SP</td>
<td>( \beta ) * Initial K-SP</td>
</tr>
<tr>
<td>C-LOGIT with fixed ( \theta ), fixed ( K_{initialSP} ) and ( \text{MaxNumberRoutes} )</td>
<td>( \beta ) * Initial K-SP</td>
<td>( \beta ) * Initial K-SP</td>
</tr>
<tr>
<td>C-LOGIT with fixed ( \text{MaxNumberRoutes} )</td>
<td>( \theta ) Initial K-SP</td>
<td>( \theta ) Initial K-SP</td>
</tr>
<tr>
<td></td>
<td>( \beta ) Initial K-SP</td>
<td>( \beta ) Initial K-SP</td>
</tr>
</tbody>
</table>

PROPORTIONAL

| Initial K-SP | \( \alpha \) Initial K-SP | \( \alpha \) Initial K-SP |
| MaxNumberRoutes | \( \alpha \) MaxNumberRoutes | \( \alpha \) MaxNumberRoutes |
| \( \alpha \) * Initial K-SP | \( \alpha \) * Initial K-SP | \( \alpha \) * Initial K-SP |
| \( \alpha \) * MaxNumberRoutes | \( \alpha \) * MaxNumberRoutes | \( \alpha \) * MaxNumberRoutes |

Table 1. Significant route choice parameters

In order to limit the number of experiments, due to the exponential growing of the number of combinations, we have fixed some parameters and taken into account only those values that fit better in the previous experiments.
Depending on the route choice model employed (proportional, logit or C-logit), the experimental design factors for the simulations were as follows:

- **Proportional route choice model:**
  - Alpha factor ($\alpha$), for which values of 0.5, 1, 2, 2.5 and 3 were considered
  - Initial K-SP, for which values of 1, 2 and 3
  - Maximum number of routes ($\text{MaxNumberRoutes}$) fixed to 3
  - Lambda factor of the cost function ($\lambda$), for which values of 0.25, 0.50 and 0.75 were considered
  If these three factors are combined, the total number of experiments is 45 ($5 \times 3 \times 3$), each of which was simulated 15 times (replications).

- **Logit route choice model:**
  - Scale factor ($\theta$), for which values of 10, 60 and 100 were considered
  - Initial K-SP, for which values of 1, 2 and 3 were considered
  - Maximum number of routes ($\text{MaxNumberRoutes}$) fixed to 3
  - Lambda factor of the cost function ($\lambda$), for which values of 0.25, 0.50 and 0.75 were considered
  If these three factors are combined, the total number of experiments is 27 ($3 \times 3 \times 3$), each of which was simulated 15 times (replications).

- **C-logit route choice model:**
  - Scale factor ($\theta$), for which values of 10, 60 and 100 were considered
  - Initial K-SP, for which values of 1, 2 and 3 were considered
  - Maximum number of routes ($\text{MaxNumberRoutes}$) fixed to 3
  - Beta ($\beta$), for which values of 0.10, 0.15, 0.50 and 1 were considered
  - Gamma ($\gamma$) fixed to 1
  - Lambda factor of the cost function ($\lambda$), for which values of 0.25, 0.50 and 0.75 were considered
  If these three factors are combined, the total number of experiments is 108 ($3 \times 3 \times 4 \times 3$), each of which was simulated 15 times (replications).

Figure 26 plots the $RGap(t)$ function and the GEH for all the experiments in which the proportional route choice model was used. In this plot, the experiments can be considered valid when the $RGap$ function is less than or equal to 10% and the GEH is greater than or equal to 80% or 85% (values based on purely empirical grounds as a rule of thumb). Therefore, the experiments accepted are located in the top, left-hand corner. If we consider this criterion in order to accept or reject the experiments, we can distinguish two separate clouds of points: one that shows an acceptable GEH and $RGap$ (to different degrees) and another that shows an unacceptable GEH and acceptable $RGap$. Acceptable GEH and $RGap$ are observed in the experiments identified in Table 2.

<table>
<thead>
<tr>
<th>Alpha factor</th>
<th>Initial K-SP</th>
<th>Lambda</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>2.5</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>0.50</td>
</tr>
<tr>
<td>2.5</td>
<td>1</td>
<td>0.50</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0.50</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0.50</td>
</tr>
<tr>
<td>2.5</td>
<td>1</td>
<td>0.75</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>0.75</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>0.75</td>
</tr>
</tbody>
</table>

Table 2 Proportional route choice model significant experiments
Figure 26: RGap and GEH of all replications using the reactive assignment procedure and the proportional route choice model

Figure 27 depicts the plot of the RGap versus GEH index of all replications using the logit route choice model. The cloud of points that are in the area of the acceptable RGap and GEH index represent 70% of the experiments in which the logit route choice was used. Table 3 identifies the experiments that have an acceptable average of RGap and GEH. It is important to highlight the fact that the scale factor fixed to 60 or 100 generates acceptable RGap and GEH, regardless of the lambda value and the initial K-SP parameter.

<table>
<thead>
<tr>
<th>Scale Factor</th>
<th>Initial K-SP</th>
<th>Lambda</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.25</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.75</td>
</tr>
</tbody>
</table>

Table 3. Acceptable RGap and GEH using a logit route choice model
Figure 27: RGAP and GEH of all replications using reactive assignment procedure and the logit route choice model.

Figure 28 depicts the plot of the RGAP versus GEH index of all replications using the C-logit route choice. The cloud of points that are in the area of the acceptable RGAP and GEH index represents 56% of the experiments in which C-logit is the route choice model used. Table 4 identifies the experiments that have an acceptable average of RGAP and GEH using a C-logit route choice model. It is important to highlight the fact that a scale factor fixed to 10 generates unacceptable RGAP and GEH, regardless of the values of the other parameters (the only combination that produces an acceptable RGAP and GEH is a scale factor fixed to 10, an initial K-SP of 1, a beta factor of 0.50 and a lambda value of 0.5). Another fact to highlight is the scale factor fixed to 60 and the Initial K-SP fixed to 1 and 2 generates an acceptable situation, regardless of the values of the beta and lambda parameters (except the combination of a scale factor of 60, an Initial K-SP of 2, a beta factor of 0.50 and a lambda value of 0.25). The same effect would be observed if the scale factor took 100 as its value, the beta factor was irrelevant, the initial K-SP was either 1 or 2 and the lambda was either 0.50 or 0.75.

<table>
<thead>
<tr>
<th>Scale Factor</th>
<th>Initial K-SP</th>
<th>Beta Factor</th>
<th>Lambda</th>
</tr>
</thead>
<tbody>
<tr>
<td>60</td>
<td>1</td>
<td>0.10</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.15</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.50</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>1.00</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.10</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.15</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>1.00</td>
<td>0.25</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>0.15</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.10</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.50</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1.00</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.50</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.10</td>
<td>0.25</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.50</td>
<td>0.25</td>
</tr>
<tr>
<td>10</td>
<td>1</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.10</td>
<td>0.50</td>
</tr>
<tr>
<td>----</td>
<td>----</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.15</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>1.00</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.10</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.15</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>1.00</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>0.10</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>0.15</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.10</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.15</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1.00</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.10</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.15</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.50</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>1.00</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.10</td>
<td>0.50</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.15</td>
<td>0.50</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.10</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.15</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>0.50</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>1</td>
<td>1.00</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.10</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.15</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>0.50</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>2</td>
<td>1.00</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>0.10</td>
<td>0.75</td>
</tr>
<tr>
<td>60</td>
<td>3</td>
<td>0.15</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.10</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.15</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>0.50</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>1</td>
<td>1.00</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.10</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.15</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>0.50</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>2</td>
<td>1.00</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.10</td>
<td>0.75</td>
</tr>
<tr>
<td>100</td>
<td>3</td>
<td>0.15</td>
<td>0.75</td>
</tr>
</tbody>
</table>

Table 4. Acceptable RGap and GEH using a C-logit route choice model

7. CONCLUSIONS

To appropriately approach the challenges that transport analysis poses to practitioners as a consequence of the evolution of traffic systems and related technologies, a new methodological paradigm is necessary, in which macro, meso and micro models should work together in a consistent way. This methodological framework can only be achieved combining computer and traffic modeling in a coherent way. This paper has presented a realization of such integrated framework showing how transport planning macro approaches, mesoscopic dynamic models and microscopic simulation can work together in the expected way, under the hypothesis the equilibrium, static as well as dynamic, is the reliable principle for such analysis.
With respect to microscopic simulation, assuming that “dynamic equilibrium” exists, the empirical results show that an appropriate time-varying k shortest paths calculation, in which the link costs are suitably defined; adequate stochastic route choice functions; and the use of a microscopic network loading mechanism achieve a network state that acceptably replicates the observed flows in the simulation horizon and a reasonable set of used paths between OD pairs, as the oscillations within a narrow band of the empirical RGap function indicate.

An additional result of the method explored is the guidance that the computational results provide for the calibration of the dynamic traffic assignment parameters, depending on the route choice function selected. This is based on the assumption that, as far as the assignment process described is concerned, and depending on how it is implemented, it can be associated with a heuristic carrying out of a preventive or reactive dynamic assignment. A proper route selection should lead to some degree of equilibrium, and the progress towards such equilibrium is measured in terms of the RGap function.

8. REFERENCES


J. Barceló and J. Casas, (2004), Heuristic Dynamic Assignment Based on AIMSUN Microscopic Traffic Simulator, Proceedings of TRISTAN V


M. Florian and D. Hearn, (1995), Network Equilibrium Models and Algorithms, Chapter 6 in: M.O. Ball et al., Eds., Handbooks in OR and MS, Vol.8, Elsevier Science B.V.


M. Florian, M. Mahut and N. Tremblay, 2002, Application of a Simulation-Based Dynamic Traffic Assignment Model, presented at the International Symposium on Transport Simulation, Yokohama (also
FHWA (2003), Guidelines for applying Traffic Microsimulation Software.
M. Patriksson, (1994), The traffic Assignment Problem: Models and Methods, VSP B.V.
H. Theil, 1966, Applied Economic Forecasting, North-Holland,