Leveraging Spatial Abstraction in Traffic Analysis and Forecasting with Visual Analytics

Natalia Andrienko[1], Gennady Andrienko[1], and Salvatore Rinzivillo[2]

Abstract— A spatially abstracted transportation network is a graph where nodes are territory compartments (areas in geographic space) and edges, or links, are abstract constructs, each link representing all possible paths between two neighboring areas. By applying visual analytics techniques to vehicle traffic data from different territories, we discovered that the traffic intensity (a.k.a. traffic flow or traffic flux) and the mean velocity are interrelated in a spatially abstracted transportation network in the same way as at the level of street segments. Moreover, these relationships are consistent across different levels of spatial abstraction of a physical transportation network. Graphical representations of the flux-velocity interdependencies for abstracted links have the same shape as the fundamental diagram of traffic flow through a physical street segment, which is known in transportation science. This key finding substantiates our approach to traffic analysis, forecasting, and simulation leveraging spatial abstraction.

We propose a framework in which visual analytics supports three high-level tasks, assess, forecast, and develop options, in application to vehicle traffic. These tasks can be carried out in a coherent workflow, where each next task uses the results of the previous one(s). At the 'assess' stage, vehicle trajectories are used to build a spatially abstracted transportation network and compute the traffic intensities and mean velocities on the abstracted links by time intervals. The interdependencies between the two characteristics of the links are extracted and represented by formal models, which enable the second step of the workflow, 'forecast', involving simulation of vehicle movements under various conditions. The previously derived models allow not only prediction of normal traffic flows conforming to the regular daily and weekly patterns but also simulation of traffic in extraordinary cases, such as road closures, major public events, or mass evacuation due to a disaster. Interactive visual tools support preparation of simulations and analysis of their results. When the simulation forecasts problematic situations, such as major congestions and delays, the analyst proceeds to the step 'develop options' for trying various actions aimed at situation improvement and investigating their consequences. Action execution can be imitated by interactively modifying the input of the simulation model. Specific techniques support comparisons between results of simulating different "what if" scenarios.

[1] Natalia Andrienko and Gennady Andrienko are with Fraunhofer Institute IAIS, Schloss Birlinghoven, 53757 Sankt Augustin, Germany and City University London, UK.
E-mail: {natalia|gennady}.andrienko@iais.fraunhofer.de.
[2] Salvatore Rinzivillo is with the Knowledge Discovery Laboratory, Istituto di Scienza e Tecnologie dell'Informazione, Area della Ricerca CNR, via G. Moruzzi 1, 56124 Pisa, Italy. E-mail: rinzivillo@isti.cnr.it.

1 INTRODUCTION

Data concerning vehicle traffic can be nowadays collected in unprecedented amounts owing to the advances in the sensing technologies. These data offer new opportunities for improving the understanding of traffic properties and enhancing the accuracy of models forecasting traffic situations and their evolution. However, the potential of real traffic data remains largely underexploited. By means of visual analytics methods, we performed a systematic focused study to investigate the potential opportunities offered by traffic data. We found that historical traffic data can not only enable understanding of spatio-temporal patterns of traffic flows and interdependencies between their key characteristics (speed and volume), but they can also be used for building mathematical and computational models capable of forecasting traffic situations and their developments under various conditions. This finding is a result of an evolutionary process of design, implementation, and application of a range of visual analytics methods focusing on movement analysis [2].

Visual analytics [61] is a field of research seeking for ways to synergistically combine the capabilities of computers with the power of human cognition in analyzing massive and complex data. Visual analytics develops integrated methods uniting computational processing with interactive visual interfaces that support human mental processes [40]. In our research on visual analytics of movement [2], we first developed a range of exploratory techniques. They allowed us to spot fundamental patterns of traffic flow dynamics [8] (briefly presented in the next section) that are common for different areas and spatial scales. The next step was development of interactive visual interfaces for representing these patterns by mathematical models [7]. After obtaining such models, a logical next step in our research was to try to use them for traffic forecasting. For this purpose, we devised a lightweight traffic simulation algorithm capable of using previously derived models and developed interactive visual embedding for defining initial conditions, running simulations, and analyzing the outcomes. Since simulations could be prepared and performed very fast, thus allowing interactive operation, we extended the tools with possibilities to imitate various interventions altering network properties and/or traffic

routes and investigate their impacts on the traffic situation development, including comparative analysis of various "what if" scenarios.

The contribution of our research is twofold. First, we comprehensively explored the potential opportunities offered by traffic data and demonstrated the possibilities of using them not only for analysis of past situations but also for efficient forecasting and "what if" analysis. Second, we developed an integrated visual analytics framework for traffic analysis, modeling, and forecasting, which supports a seamless workflow involving different analytical tasks. Specifically, our framework supports three major types of analytical tasks ([61], p. 35): assess (i.e., understand the piece of reality represented by available data), forecast (i.e., estimate the properties or behavior of the piece of reality beyond the part represented in the data), and develop options (i.e., find actions that can change the properties or behavior of the piece of reality in a desired way).

Although the three types of tasks can be treated as independent activities [61], it is obvious that these can also be stages of a single analytical process, in which 'forecast' builds on results of 'assess' and, in turn, enables 'develop options'. So far, it has not been usual for visual analytics researchers to strive at supporting the whole process comprising all three types of tasks. Our work thus makes a contribution to visual analytics research not only by proposing a set of application-oriented techniques but also by demonstrating a way to support such a process. Although the immediate results of our work are specific to traffic analysis, the experience we gained and the framework incorporating it can be generalized and used by other researchers. The flow chart in Fig. 1 schematically represents a generalized view of an analytical process and exhibits the generic subtasks that need to be supported by visual analytics methods.

[Fig. 1. Three types of visual analytics tasks as stages of a single analytical process.]

At the first stage ('assess'), analysts study available data to understand the phenomena reflected in the data. The understanding gained by the analysts can be seen as a mental model of the phenomenon. It is beneficial for the further analysis to externalize this model, preferably, in a form permitting computer processing. The analysts may derive several models representing different aspects of the phenomenon. Each model must be assessed and validated. At the second stage ('forecast'), the models are used to forecast events and developments that can occur under some conditions of interest. The analysts assess the resulting forecast to see the "capabilities, threats, vulnerabilities, and opportunities" ([61], p. 35) and understand whether the predicted situation or process requires intervention. If so, the analysts proceed to the third stage ('develop options'), in which they choose actions for improving the situation/process and perform the task 'forecast' to see and assess the effects of the chosen actions. The analysts also need to compare predicted results of different possible interventions.

This general scheme can be instantiated for various applications. The generic subtasks need to be specialized for a given application, thereby setting specific objectives for design and development of visual analytics tools.

In this paper, we present an instantiation of the general scheme for vehicle traffic analysis and include a special discussion of possible ways to validate traffic forecasting models using real data. Due to the novelty of our approach to traffic modeling and simulation, we find it reasonable to begin with presenting evidence of the existence and uniformity of the fundamental traffic relationships in spatially abstracted transportation networks at different spatial scales (section 2). After this, we give an overview of related works (section 3) and then describe our framework in section 4, dealing consecutively with the tasks 'assess' (analysis and modelling of historical traffic data; subsection 4.1), 'forecast' (traffic simulation; subsection 4.2), and 'develop options' (forecasting and comparison of consequences of possible traffic regulation actions; subsection 4.3). Section 5 presents our approach to the evaluation and validation of models built according to our framework, and section 6 contains a concluding discussion. Additionally, the web page http://geoanalytics.net/and/is2015/ contains full color versions of all figures, a video demonstrating the analytical process, and a slide presentation.

2 LEVERAGING SPATIAL ABSTRACTION: APPROACH SUBSTANTIATION

2.1 Creation of a spatially abstracted transportation network

A *trajectory* of a moving object consists of records reporting the positions (e.g., geographic coordinates) of the object at different times. Given a large set of trajectories, we apply a previously developed method [5] that derives an abstracted network consisting of territory compartments (further called *cells*) and abstract *links* between them. In brief, the method first organizes points sampled from the trajectories into groups such that each group fits in a circle of a user-specified maximal radius (e.g., 1 km); the actual radius of a group may also be smaller. The medoid of each group (i.e., the point with the smallest sum of distances to all other points) is taken as a generating seed for Voronoi tessellation; hence, the resulting cells are Voronoi polygons. Smaller or larger cells can be generated by varying the maximal circle radius, thus allowing traffic analysis at a chosen spatial scale. Moreover, it is also possible to vary the spatial scale across the territory depending on the data density and thus obtain finer cells in data-dense areas and coarser cells in data-sparse regions [2].

After obtaining a division of the underlying territory into cells, the trajectories are transformed into *flows* (aggregate movements) between the cells. For each ordered pair of cells C_i and C_j, a *flow* is an aggregate of individual moves from C_i to C_j, where a *move* is a pair of consecutive points from one trajectory such that the first point is inside C_i and the second point is inside C_j. Two cells C_i and C_j are said to be *linked* if there is at least one move from C_i to C_j in the dataset. An ordered pair of linked cells is called a *link*. Note that such a link is not a physically existing entity but an abstract construct. A combination of a set of cells C and a set of links L is considered as an *abstract transportation network*. In mathematical terms, <C, L> is a graph where C is the set of nodes and L is the set of edges.

As an illustration, Figure 2 shows a map with a spatially abstracted transportation network of Milan (Italy) reconstructed from GPS tracks of 17,241 cars collected over a period of one week from Sunday, April 1, to Saturday, April 7, 2007 (data source: Octo Telematics SpA; http://www.octotelematics.com). The territory of Milan is divided into cells with approximate radii of 1 km. The cell boundaries are shown as grey lines. For showing the abstract links, we apply the flow map technique [43][59], which represents flows by straight or curved linear symbols. The widths and/or colors of the lines may encode characteristics of the flows. Movement directions can be shown by arrows; however, when it is necessary to represent opposite flows between the same places, cartographers and visualization designers use half-arrows [62] or curved lines with the curvature increasing towards the flow destination [67]. These two possibilities are illustrated in enlarged map fragments on the right of Fig. 2. In the whole map, curved lines are used. For improving the map legibility, the flow symbols are differently colored. The meaning of the colors will be explained later.

[Fig. 2. A spatially abstracted transportation network of Milan (Italy) with cell radii ≈ 1km.]

2.2 Spatio-temporal aggregation of movement data

For the further analysis, the trajectories are aggregated into flows by time intervals. For each link (C_i, C_j) and each time interval T_k, the corresponding flow is an aggregate of all moves from C_i to C_j that ended within the interval T_k and started within either T_k or T_{k-1}. The flow is characterized by the number of moves and the mean speed (velocity) of the movement. The number of moves (traffic volume) per time interval is called *traffic intensity* (in some literature, the terms *traffic flow* or *flux* are used). The mean speed is computed as follows. For each object that moved from cell C_i to cell C_j, two trajectory points that are the closest to the centers of these cells are selected. Dividing the length of the path between the selected points by the time difference between them gives the mean speed of this object. The overall mean speed on the link (C_i, C_j) in a time interval T_k is computed as the mean of the mean speeds of all objects that moved from C_i to C_j during T_k. Hence, the aggregation of the trajectories into flows by time intervals produces two sets of time series associated the links: traffic intensities and mean speeds.

In our example in Fig. 2, we applied partition-based clustering (k-means) to the time series of flow characteristics, i.e., the links have been clustered by the similarity of the associated time series of the flow intensities and mean speeds. Colors have been assigned to the resulting clusters in such a way that similar colors correspond to similar clusters [7]. These colors have been used to paint the flow symbols on the map. The color legend on the left of Fig. 2 shows the colors assigned to the clusters and the cluster sizes, i.e., the number of links in each cluster and the percentage of the total number of links.

2.3 Interdependencies between traffic intensities and mean speeds

To study and quantify the relationships between the traffic intensities and mean speeds on the links, the data are transformed in the following way. Let A and B be two time-dependent attributes associated with the same object (in particular, link) and defined for the same time steps. In particular, A may stand for the traffic intensity and B for the mean speed, or vice versa.

1. Divide the value range of attribute A into intervals.
2. For each value interval of A:
   a. Find all time steps in which the values of A fit in this interval.
   b. Collect all values of B occurring in these time steps.
   c. From the collected values of B, compute summary statistics: quartiles, 9th decile, and maximum.
   d. For each statistical measure, construct a sequence of values of B corresponding to the value intervals of A.

In this way, a family of attributes is derived: first quartile of B, second quartile (median) of B, third quartile of B, 9th decile of B, and maximum of B. Thus, if A stands for the traffic intensity and B for the mean speed, the derived attributes are: first quartile of the mean speed, median of the mean speed, and so on. For each of the derived attributes, there is a sequence of values corresponding to the chosen value intervals of attribute A, for example, a sequence of aggregate speed values for different intervals of the traffic intensity. This sequence is

similar to a time series except that the steps are based not on time but on values of attribute A. We call such sequences dependency series since they express the dependency between attributes A and B. Attribute A is treated as the independent variable and B as the dependent variable.

To study and model the interdependencies between the mean speed and the traffic intensity, we perform two transformations. First, we treat the traffic intensity as the independent variable and derive a family of attributes expressing the dependency of the mean speed on the traffic intensity. Second, we treat the mean speed as the independent variable and derive a family of attributes expressing the dependency of the traffic intensity on the mean speed. Dependency series may be derived using either the absolute or relative traffic intensities, the latter being computed as the ratios or percentages of the absolute intensities to the maximal intensities attained on the same links.

[Fig. 3. The graphs represent the interdependencies between the traffic intensity and mean speed for the link of the abstracted transportation network of Milan shown in Fig. 2.]

The dependencies we have derived for the abstracted transportation network of Milan shown in Fig.2 are graphically represented in Fig.3. The lines in the graphs correspond to the links of the network and are colored according to the cluster membership of the links using the same colors as in Fig.2. The graph on the left represents the dependencies of the mean speed on the relative traffic intensity in percent to the maximum. The horizontal axis corresponds to the traffic intensity (N moves % of max) and the vertical axis to the 9th decile of the mean speed. We have taken the 9th decile because this statistical measure is less sensitive to outliers as the maximum. Outliers among the values of the mean speed often occur in time intervals of low traffic intensity, when a single or only a few vehicles traverse a link. The graph on the right represents the dependencies of the relative traffic intensity on the mean speed. The horizontal axis corresponds to the mean speed and the vertical axis to the maximal traffic intensity.

On the left of Fig.3, the shapes of the lines show that the mean speed decreases with increasing traffic intensity. On the right, the lines have the shape of a bell or symbol '$\cap$'. This shape can be interpreted as follows. When vehicles move with a low mean speed, only a small number of vehicles can traverse a link in a time unit. When the mean speed increases, the number of vehicles also increases, but only till the point when a certain "optimal" value of the mean speed is reached. After this point, movement with higher mean speeds is only possible when the traffic load decreases. These observations conform to our commonsense knowledge and experiences concerning the behavior of the vehicle traffic on roads but refer to an abstracted rather than physical transportation network.

[Fig. 4. The dependencies between the traffic intensity and mean speed can be represented by polynomial regression models.]

Figure 4 demonstrates how the dependencies the traffic intensity and mean speed can be represented by formal models, such as polynomial regression (other kinds of curves can be fitted as well). The modeling is done for clusters of links rather than for each individual link, to avoid over-fitting and reduce the impact of local outliers and fluctuations. The fitted curves capture the character of the dependencies. These curves have the same shape as the fundamental diagram of traffic flow describing the relationship between the flow velocity and traffic flux (i.e., intensity) [27] (see also http://en.wikipedia.org/wiki/Fundamental_diagram_of_traffic_flow). The fundamental diagram refers to links of a physical transportation network, i.e., to street segments. The exact parameters of the curves depend on the street properties, such as the width, number of lanes, and speed limit. We see that the same relationships as in a physical network exist also in a spatially abstracted network. Moreover, we have found that the relationships conforming to the fundamental traffic diagram exist on different levels of spatial abstraction, as illustrated in Fig. 5. The parameters of the curves depend on the properties of the abstracted links. As each abstracted link stands for a group of physical links, its properties incorporate and summarize the properties of these physical links.

[Fig. 5. The maps show spatially abstracted transportation networks of Milan with cell radii ≈ 2km (top) and 4 km (bottom). The graphs to the right of each map represent the dependencies between the relative traffic intensities and the mean speeds on the network links.]

For the Milan data, we have checked our finding for abstract networks built with the following values of the parameter 'maximal radius': 750 m, 1 km, 1.5 km, 3 km, 4 km, 4.5 km, and 5 km. We have also checked it for a much larger dataset covering the geographical region of Tuscany (Italy) and a time period of one month. Similar relationships as in Milan have been observed at diverse spatial scales for traffic flows both within and between the towns of Tuscany. This key finding provides a basis for our approach to traffic analysis and modelling.

2.4 Advantages and limitations of spatial abstraction

The fundamental relationships between the traffic flow characteristics represented by the conventional traffic flow diagram are commonly used for traffic flow prediction and simulation, which is usually done for a physical street network. The existence of the same relationships at higher levels of spatial abstraction makes it possible to do modeling, prediction, and simulation also at higher spatial scales in cases when fine details are not necessary. Spatial abstraction of a street network offers the following advantages:

- The number of nodes and links in an abstracted network can be much smaller than in the underlying physical network. Hence, much less time and effort is needed for model building and calibration, and also simulations can be carried out much faster compared to the current practices. This enables, in particular, rapid approximate predictions and assessments in emergency situations, when time is very limited.
- Spatial abstraction compensates for the sparseness of real data on streets with low traffic. There may be not enough trajectory points on a given street segment for reconstructing the dependency between the mean speed and traffic intensity, but aggregation of several physical links into one abstract link alleviates the problem.
- As mentioned earlier and will be shown by example further in the paper, it is possible to build an abstract network in which the level of spatial abstraction varies across a territory according to the variation of the data density. In areas with high traffic, abstracted links may very closely approximate physical links (i.e., street segments), whereas areas with low traffic can be represented by large cells. Hence, it is possible to have different levels of detail in traffic simulations and prediction in areas with high and low traffic, when fine details in low traffic areas are not important.
- Spatial abstraction may also serve as a tool for protecting locational data privacy [50].

There are certain limitations for applying spatial abstraction. Obviously, it is not applicable to problems requiring detailed analysis and modelling of traffic at the level of street segments and junctions. However, even when problem settings permit abstraction, the spatial scale (i.e., the cell sizes) cannot be unlimitedly increased without distorting and eventually destroying the shapes of the curves representing the relationships between the traffic fluxes and velocities. Generally, increasing the spatial scale increases the amount of noise (i.e., oscillations) within the curves. The overall shapes of the curves remain discernible up to a certain abstraction level, at which the oscillations become too high. Our experiments show that the upper limit for the cell sizes may depend on the number and diversity of the existing physical links between the cells. Thus, for Milan and the urban areas of Tuscany, increasing the cell radius beyond 4 km distorts the curves too much, whereas much larger cells can be used for the rural areas of Tuscany. Hence, there is no uniform upper limit to the level of spatial abstraction that would be valid everywhere. An appropriate level for a given territory and available data can be determined empirically with the use of visual analytics techniques.

One reservation needs to be made concerning the reconstruction of the fundamental relationships between the traffic flow characteristics from real vehicle trajectories. It is typical that available trajectories cover only a sample of vehicles that move within a network and not the entire population. Hence, the traffic intensities computed from these trajectories need to be appropriately scaled, to approximate the real intensities. This reservation is not specific to spatially abstracted networks but also applies to detailed street networks. Appropriate scaling parameters (or even scaling functions capturing daily and weekly variations) can be obtained by comparing the vehicle counts computed from trajectory data with real traffic counts obtained from traffic sensors [54]. The issue of data scaling lies out of the scope of this paper.

3 RELATED WORKS

3.1 Visual Analytics Methods for Analysis of Movement Data

Analysis of movement data, including data concerning vehicle traffic, is one of the most important topics in many fields of research, including data mining [29], geographic information science [31], and visual analytics [6]. The majority of the existing works concentrate on (1) visualization and interactive exploration of spatial and temporal aspects of individual trajectories [39][63] and sets of trajectories [32][36], (2) analysis of movement attributes along trajectories [32][60][63][68], (3) detecting stops, interactions between trajectories, and other kinds of events [9][10], (4) aggregating movement data in space and time and visualization of resulting aggregates [23][65][66][67], and (5) revealing relationships between movement and the environment (context) [45].

Generally, while there exists a wide variety of visual analytics methods oriented to movement data [6] that can support the ‘assess’ task in movement analysis, the tasks ‘forecast’ and ‘develop options’ have been barely addressed. One work [7] proposes a visual analytics framework to support representing spatio-temporal patterns of traffic flows and relationships between flow characteristics by formal models. Such models can, obviously, be used for forecasting, but the transition from ‘assess’ to ‘forecast’ and ‘develop options’ is not supported.

Works of Sewall et al. [56][57] focus on the 'forecast' task in movement (specifically, vehicle traffic) analysis. They developed traffic simulation algorithms allowing generation of realistic 3D animations of simulated vehicle movements. However, the forecasting is not linked to previous analysis of real data and to following development of options.

3.2 Visual Analytics Support to Modeling and Simulation

Multiple works in visual analytics support building and assessment of models in other application domains. Much attention has been given to classification models [25][30][69], including decision trees [24]. A classifier may be built for assigning new objects to previously obtained clusters [3][26]. Visual analytics techniques also support derivation of linear trend models from multivariate data [32], regression models [51], and time series models [13][34]. Other works focus on supporting assessment of existing models, including classification [49], engineering simulation [47][48], and spatio-temporal models [21][46][55]. In particular, comparative assessments of results of multiple simulations are supported [47][48][64].

There are works on supporting the tasks 'forecast' and 'develop options' in planning pandemic response [1][46] and flood protection [55][64]. Pre-existing simulation models are used to provide forecasts. Interactive environments allow users to assess the forecasts, imitate implementation of response measures, observe the effects of these measures, and compare expected results of different measures. The system World Lines [64] represents multiple simulation runs in a tree-like view. Each simulation is shown as a branch. User's interventions create new branches, which are executed in parallel with previously existing ones allowing the user to compare the consequences of different response measures. In RunWatchers [41], a response plan is created automatically using multiple parallel simulations, and then the system visualizes the generated decision tree and presents the resulting plan in the form of a visual storyboard representing the recommended sequence of actions and justifications for the decisions taken.

We are not aware of visual analytics works on supporting the whole workflow 'assess' – 'forecast' – 'develop options' as presented in the introduction.

3.3 Transportation Research

According to the level of detail in representing traffic, there exist macroscopic, mesoscopic, and microscopic traffic simulation models [57]. Macroscopic models (a.k.a. continuous or continuum models [56]) describe the traffic at a high level of aggregation as flow without considering individual vehicles. This is done using sets of differential equations, which are often defined based on analogies to physical phenomena, such as gas or fluid dynamics [44]. The cell transmission model [18][19] is a discretized variant of the macroscopic approach.

In microscopic models, traffic is described at the level of individual vehicles and their interactions with each other and with the road infrastructure. Two major classes are agent-based models [53] and cellular automata models [52]. Being quite resource-demanding, microscopic models have traditionally been used for local simulations in small areas, but the increased power of computers and parallel computing have enabled microscopic simulations for large networks. A disadvantage of microscopic models is large effort required for model preparation.

Mesoscopic models fill the gap between macroscopic and microscopic models by combining individual vehicle representation with aggregate representation of traffic dynamics [16]. Individual vehicles or packets of vehicles move through links of a transportation network according to general speed-density relationships defined in traffic flow theories [20][27] or derived from real data [35]. These relationships involve a number of parameters, which can be set differently for different link types [16].

Hybrid simulation models combine macroscopic or mesoscopic models with microscopic models [14][15]. Different model types are applied to different parts of a network. Thus, Sewall et al. [57] perform agent-based simulation of individual vehicles in regions of user's interest while a faster macroscopic model is used in the remainder of the network. Our algorithm, which is presented in section 4.2.1, can be classified as a hybrid of mesoscopic and microscopic simulation methods: it simulates in detail only movements of additional vehicles while regular traffic flows are involved in the simulation in an aggregate form.

There are a number of commercial and open-source software packages for traffic simulations [38][42]. An open-source package SUMO [12] is used for preparing and performing micro- and macroscopic simulations. In our framework, however, we do not use any of these packages because the simulation models they include cannot be easily fed by the outputs of the 'assess' stage.

3.4. How this Work Extends the State of the Art

With regard to the visual analytics research on analysis of movement data, our work goes beyond supporting only the 'assess' task, as in almost all existing works, and extends the research scope to the tasks 'forecast' and

'develop options'. While the task 'forecast' alone has been addressed by Sewall et al. [56][57], our work is the first to propose visual analytics support to all three tasks together in a common framework.

Moreover, in the visual analytics research as a whole, it is usual to address one of the tasks, and only a few works address two of them [1][46][55][64]. Our work gives a first example of how all three tasks can be supported as a single workflow. It also proposes a general scheme of this workflow, which can be used in developing comprehensive visual analytics support for other applications.

In the transportation research, a usual way to use real data is for setting parameters of pre-existing theory-based models. Another usual feature is performing traffic simulation at the level of detailed street network. Distinctive features of our approach are the use of spatial abstractions of physical transportation networks and derivation of models for traffic forecasting (simulation) directly from real historical data.

4 PRESENTATION OF THE FRAMEWORK

We have developed a visual analytics framework supporting the task workflow 'assess' – 'forecast' – 'develop options' in analysis of network-constrained movement in geographic space. The framework is applicable to aggregated movement data associated with nodes and links of a network, which may be a physical street network or a spatially abstracted network, as described in section 2. The data must include traffic intensities, that is, counts of objects that moved through the links by time intervals, and corresponding mean speeds of their movement. Ideally, the counts should represent the entire population of the objects that moved over the network, but the framework can also be applied to counts representing a large sample of objects (see the reservation at the end of section 2). The length and temporal resolution of the time series must be suitable for capturing the traffic variation related to the daily and weekly temporal cycles. This means that the length must be at least one week (more is better) and the resolution must be at most one hour (finer is better); a coarser temporal resolution may conceal important daily patterns, such as morning and evening rush hours.

In describing the framework, we use a running example of aggregated data from the area of Tuscany in Italy. The data were derived from about 3 million GPS tracks (i.e., sequences of position records, more than 52 million records in total) of 42,686 private cars collected during May 2011 by company Octo Telematics SpA (http://www.octotelematics.com). To protect the personal privacy of the car owners, access to the original data could not be provided, but we could obtain spatially and temporally aggregated data. For this purpose, the territory was divided into compartments (Voronoi cells) [2][5] based on a random sample of 90,000 position records. The sizes of the cells vary across the territory depending on the data density. The original data were aggregated by these cells and 744 hourly time intervals from May 1 to May 31, 2011.

4.1 Assess

The visual analytics support to the 'assess' task includes techniques for exploratory analysis of movement data and interactive tools for derivation of formal models representing the spatio-temporal variation of traffic flows and the relationships between the traffic intensities and mean speeds [2][7]. The process of model building is supported by an interactive visual interface illustrated in Fig. 4. It allows the user to choose a suitable method from a library of modeling methods, test different parameter settings and find appropriate ones, generate a model, evaluate its quality, and, when necessary, refine the model.

In our approach, models are built for clusters of links of a (spatially abstracted) transportation network rather than for individual links, to avoid over-fitting and reduce the impacts of noise and local outliers. The links are clustered according to the similarity of the associated time series (TS) of traffic intensities and mean speeds using a partition-based clustering algorithm, such as k-means, and interactive tools enabling visually supported progressive clustering [7]. We build three sets of models: (1) models of the temporal variation of the traffic intensity in each cluster, (2) models of the dependencies of the mean speeds on the traffic intensities, and (3) models of the dependencies of the traffic intensities on the mean speeds. For the model set (1), we apply the double exponential smoothing (Holt-Winters) method [36], which captures the periodic character of the temporal variation regarding the daily and weekly time cycles. For the model sets (2) and (3), we apply polynomial regression models, as demonstrated in Fig.4.

Each model by itself makes a common prediction for all cluster members, but this prediction is individually adjusted for each cluster member based on the statistics of the distribution of its original values [7].

For our running example, the graph in Fig. 6A demonstrates the dependency series (DS) of the 9th decile of the mean speed depending on the relative traffic intensity. The lines are colored according to the cluster membership of the links; the spatial distribution of the clusters is shown in Fig. 7. In Fig. 6B, the dependencies are represented by polynomial or linear regression models derived for each cluster. For most clusters, polynomial regression models adequately represent the dependency. For clusters characterized by very low traffic intensities, linear regression models are more suitable. It can be seen that all curves have almost the same

shape (which means the same character of the dependency) and differ only in the level on the Y-axis, i.e., the speed values attained.

[Fig. 6. A: The dependency series of the mean speed versus the traffic intensity have been clustered by similarity. B: The dependencies are represented by polynomial or linear regression models.]

[Fig. 7. The links in the flow map are colored according to the cluster membership of the dependencies of the mean speeds on the traffic intensities (Fig. 6).]

Fig. 8 demonstrates the modeling of the opposite dependency: how the maximal number of vehicles that can pass through a link during one hour depends on the mean speed. As previously, the DS of the maximal traffic intensities depending on the mean speeds have been clustered by similarity. The graph Fig. 8A shows DS from three selected clusters (when all clusters are shown simultaneously, the display gets too cluttered and illegible). The map in Fig. 9 shows the spatial distribution of all link clusters. To capture the dependencies, we apply polynomial regression models with higher polynomial orders than for the previous set of models, since the line shapes are more complex. Fig. 8B graphically represents the dependency models built for all clusters.

[Fig. 8. A: Three selected clusters of dependency series of the traffic intensity depending on the mean speed. B: The dependency curves built for all clusters.]

[Fig. 9. The links are colored according to the cluster membership of the dependencies of the traffic intensities on the mean speeds.]

To summarize, the 'assess' stage of the traffic analysis workflow is supported by interactive tools for progressive clustering (allowing refinement of selected clusters), data visualization on maps and graphs, interactive visual interface to a modeling library, and techniques for model assessment by analyzing residuals [7]. The output of the 'assess' stage consists of three sets of models:

- **TMF** (Temporal Models of Flows): a set of models predicting for each link the regular number of moving vehicles (traffic intensity) in different time intervals;
- **DMSL** (Dependency Models of Speed on Load): a set of models predicting the mean speed of the movement through each link depending on the load, i.e., the number of vehicles that try to move;
- **DMFS** (Dependency Models of Flow on Speed): a set of models predicting the maximal number of vehicles that will be able to move through each link depending on the mean speed with which the vehicles can move.

Model descriptions are stored externally in XML files. Based on these descriptions, the models can be later restored and used for traffic forecasting.

4.2 Forecast

Assuming that TMF appropriately represents the daily and weekly variation of the usual traffic, this set of models can be directly used for forecasting the usual traffic in a time period that is not covered by the available data. This kind of forecasting is supported by a tool that allows the user to specify the time period for which the forecast needs to be done. The tool divides the period into time intervals of the same length as in the description of TMF and then applies the model set to each link and each time interval to forecast the expected traffic intensity.

DMSL and DMFS are used for simulating unusual extra traffic that appears in addition to the regular traffic. We do not use any of the existing traffic simulation tools for two main reasons: (1) high effort required for representing the network topology and link properties, and (2) absence of an easy way to feed the tool with the dependencies defined by DMSL and DMFS. To enable a seamless transition from 'assess' to 'forecast' and the use of simulation in interactive settings (for this purpose, it needs to be fast), we have developed our own traffic simulation algorithm that can directly utilize all three sets of models derived from real data. No effort for network encoding and parameter setting is required, which allows the user to proceed seamlessly from the task 'assess' to the task 'forecast'. The algorithm is implemented within the software system V-Analytics, which is downloadable from http://geoanalytics.net/V-Analytics/ and may be used free of charge for research and educational purposes.

4.2.1 Traffic Simulation Algorithm

As mentioned in section 3.3, our algorithm simulates in detail only movements of additional vehicles while regular traffic flows, which are predicted by TMF, are involved in the simulation in an aggregate form. To our knowledge, this is a novel way to simulate traffic. Traditionally, hybrid models use different model types for

different parts of the network [14][15], while we use them for different parts of the traffic, regular and extraordinary. Our approach allows the user to study specifically the development of the extra traffic and its interactions with the regular traffic. Still, the method can also be used in a more traditional form. It can simulate the movement of an explicitly specified population of vehicles without involving any background traffic flows, that is, assuming that the given population includes all vehicles existing on the territory under study. For this task, the simulation method does not use TMF and uses only DMSL and DMFS.

The main idea is following: for each link, the method finds how many vehicles need to move through it in the current minute, determines the mean speed that is possible for this link load (using DMSL), then determines how many vehicles will actually be able to move through the link in this minute (using DMFS), and then promotes this number of vehicles to the end place of the link and suspends the remaining vehicles in the start place of the link. A formal description of the algorithm is given below.

The system of places and links is represented as a directed graph G = (P, L), where P is the set of places and L is the set of links. Each link has a collection of dynamic (time-variant) properties. Current properties of the links can be determined by querying the models TMF, DMSL, and DMFS:

- TMF(l, t) gives the expected number of regular vehicles traversing link l at time t;
- DMSL(l, n) gives the expected mean speed of n vehicles moving on link l;
- DMFS(l, s) gives the maximal number of vehicles able to move on link l with mean speed s.

The input of the simulation procedure includes the graph G, the models, and a population of additional vehicles V, where each vehicle has been assigned an origin node (place), a destination node, a route, i.e., a path through the graph from the origin to the destination, and a time moment when the vehicle can start the movement.

At each time instant t, a vehicle can be in one of the following states (Fig. 10):

- SUSPENDED: the vehicle is located in a node and is waiting to move towards the next node in its planned route;
- MOVING: the vehicle is moving on a link to reach the next node of its planned route;
- ARRIVING: the vehicle arrives at one of the nodes of its planned route.

[Fig. 10. The possible states of vehicles and transitions between the states in the course of the simulation.]

If the current conditions allow a suspended vehicle to move, it can pass to the status MOVING; otherwise, it remains in the SUSPENDED status. When a vehicle changes the status to MOVING, it is assigned an expected time of arrival to the next node; the current node, in which the vehicle has been till this moment, is removed from the route of this vehicle. The vehicle will remain in the MOVING status until the simulation time reaches the assigned arrival time. When this happens, an arrival event is emitted reporting the identifier of vehicle, the current node to which the vehicle has arrived, and the arrival time. The set of emitted arrival events allows reconstructing the movement history (i.e., the trajectory) of each vehicle. If the current node is the final destination of the vehicle, the vehicle is removed from the simulation; otherwise, the status of the vehicle is turned to SUSPENDED. Generally, the following relationships between the current simulation time t, the arrival time of a vehicle v, and the status of the vehicle v hold:

- if v.arrival_time < t then v.status = MOVING;
- if v.arrival_time = t then v.status = ARRIVING;
- if v.arrival_time > t then v.status = SUSPENDED.

The assigned time of the movement start of each vehicle is treated as the time of its arrival to the first node of its route, i.e., the origin place of the trip. If the assigned movement start time is later than the simulation start time, the vehicle will be treated as MOVING towards its designated origin place until the simulation time reaches the movement start time.

At each stage of the simulation, i.e., at time t, the following procedure is performed:

1. Emit arrival events for all vehicles in ARRIVING status (MOVING → ARRIVING).
   For each *vehicle* ∈ V:
     if *vehicle.arrival_time* = *t* **then**
       emit(*vehicle*, head(*vehicle.route*), *t*) ;
       if |*vehicle.route*| = 1 **then**
       // *vehicle* has reached final destination
         let V = V – { *vehicle*} // remove *vehicle* from V
2. **For each** *link* ∈ L, *link* = (p_i, p_j), p_i ∈ P, p_j ∈ P:
   2.1. Determine current link properties:
     let *SuspendedVehicles* =
       {*vehicle* ∈ V | *vehicle.arrival_time* < *t* and

vehicle.route is like [p_i, p_j,..., p_n] };
let *link.baseLoad* = TMF(*link*, *t*) + *link.restBaseLoad*;
let *totalLoad* = *link.baseLoad* + |*SuspendedVehicles*|;
let *meanSpeed* = DMSL(*link*, *totalLoad*);
let *maxFlow* = DMFS(*link*, *meanSpeed*);
let *traversalTime* = *link.pathLength* / *meanSpeed*

2.2. Select vehicles to pass to next node p_j:
let *ratioCanPass* = *maxFlow* / *totalLoad*;
let *nBaseCanPass* = round (*ratioCanPass***link.baseLoad*);
let *nExtraCanPass* = min(*maxFlow* – *nBaseCanPass,* |*SuspendedVehicles*|);
let *VehiclesToPass* = subset(*SuspendedVehicles*, *nExtraCanPass*);
let *VehiclesToWait* = *SuspendedVehicles* – *VehiclesToPass*;
let *link.restBaseLoad* = *link.baseLoad* – *nBaseCanPass*

2.3. Update the statuses of the vehicles:
For each *vehicle* ∈ *VehiclesToPass*:
let *vehicle.route* = tail(*vehicle.route*)
/ / remove the first element of the sequence
let *vehicle.arrival_time* = *t* + *traversalTime*
let *vehicle.waitingTime* = *0*
For each *vehicle* ∈ *VehiclesToWait*:
let *vehicle.waitingTime* = *vehicle.waitingTime* + 1

3. Propagate the base traffic suspensions remaining on the links back to their adjacent links.
For each *link* ∈ L, *link* = (p_i, p_j), p_i ∈ P, p_j ∈ P:
if *link.restBaseLoad* > 0 **then**
let *InLinks* = { *inLink* ∈ L | *inLink* = (p_k, p_i) };
/ / set of incoming links to place p_i
for each *inLink* ∈ *InLinks*, *inLink* = (p_k, p_i)
let *inLink.weight* = *inLink.baseLoad* * distance (p_k, p_j);
let *sumWeights* = Σ (*inLink.weight* | *inLink* ∈ *InLinks*);
/ / Distribute *link.restBaseLoad* among *InLinks*
/ / proportionally to their weights
for each *inLink* ∈ *InLinks*
let *inLink.restBaseLoad* = round
(*link.restBaseLoad* * *inLink.weight* /*sumWeights*);
let *link.restBaseLoad* = 0

4. (Optional step) Find alternative routes for vehicles that have been suspended for long time.
For each *vehicle* ∈ V:
if *vehicle.waitingTime* > *vehicle*.max*WaitingTime* **then**
let *vehicle.route* = find_fastest_path
(head(*vehicle.route),* last(*vehicle.route*),G)

Steps 1 and 2 constitute the basic simulation model. Step 3 extends the basic model by propagating traffic retardations over the network. When an outgoing link from a place is so overloaded that some vehicles cannot pass, the incoming flows to this place slow down because there is not enough room in the place to accommodate the arriving traffic. To model this impact, the rest load suspended on the outgoing link is distributed among the incoming links. The additional loads will retard the traffic on these links in the next simulation step.

For distributing the rest load over the incoming links, these links are assigned weights based on the following assumptions: (a) incoming links with higher loads are affected more by the traffic retardation on the outgoing link, and (b) vehicles strive to follow possibly straight paths avoiding sharp turns and returns to earlier visited places. Accordingly, the weights are the products of the loads of the incoming links (accounting for assumption (a)) and the distances from their origins to the destination of the outgoing link (accounting for assumption (b)). Indeed, the sharper the angle between two links is, the shorter is the distance between the origin of the incoming link and the destination of the outgoing link, which decreases the weight. For two consecutive links lying on a straight line, the distance between the origin of the first link and the destination of the second link is maximal. For two opposite links (i.e., when the origin of the incoming link coincides with the destination of the outgoing link), this distance is zero, which turns the weight also to zero.

Step 4 introduces dynamic re-routing of vehicles. When a vehicle has been suspended for a long time, it may decide to try to go to its destination by another route. A new fastest route from the current location of the vehicle to its destination is computed taking into account the times required for passing the links at the current traffic conditions. For each link, the required passing time is computed as the sum of the average link traversal time and the average vehicle waiting time on this link in the past k simulation steps.

The algorithm terminates when all vehicles have arrived to their final destinations (i.e., the set V becomes empty) and no rest load remains on any link. The user can limit the time period for which the simulation needs to be done. In this case, the algorithm stops when the simulation time t reaches the end of this period. The arrival events emitted by the algorithm are transformed into trajectories of the vehicles.

4.2.2 Simulation Algorithm Complexity

Before the simulation, each vehicle needs to be assigned a route from its origin to its destination. The complexity for determining the fastest route between two nodes of a directed graph G=(P,L) using Dijkstra's algorithm [22] is $O(|L|+|P|\cdot\log|P|)$. For n vehicles, the overall complexity is $O(n\cdot(|L|+|P|\cdot\log|P|))$. It can be amortized by computing all routes in advance and storing them for fast retrieval.

The complexity of the simulation algorithm itself depends on the complexity of one simulation move consisting of steps 1-4. In step 1, the status of n vehicles is checked, resulting in O(n). In step 2, the set of vehicles suspended on each link is determined. We use a data structure in which each vehicle is attached to the next node of its planned route. In this case, the cost of finding all suspended vehicles for a link is constant (the vehicles are attached to the link origin) and the overall complexity of step 2 is $O(|L|)$. In step 3, the condition of *baseRestLoad* is evaluated for each link, thus requiring a complexity of $O(|L|)$. Step 4 requires the computation of new routes for vehicles that are waiting for a long time in congested nodes, i.e. a cost of $O(|L|+|P|\cdot\log|P|)$ for each such vehicle. In the worst case, it may take time $O(n\cdot(|L|+|P|\cdot\log|P|))$, but it is very unlikely that all vehicles in all nodes will simultaneously need re-routing. In practice, only a small fraction of all vehicles will require this step. Thus, in our simulations, there were only a few critical places where many vehicles had to wait for long time. Therefore, we can assume a cost of $O(|L|+|P|\cdot\log|P|)$ for each time step.

In summary, assuming to have pre-computed the initial routes for all vehicles, the resulting complexity of T time steps is $O(T\cdot(n+|L|+|P|\cdot\log|P|))$. If the dynamic re-routing (step 4) is omitted, the complexity is $O(T\cdot(n+|L|))$. In our examples, with the algorithm implemented in Java and running on a commodity desktop computer, the simulation of movements of 1000 vehicles during about 10 hours takes less than 20 seconds, which is reasonable for using the algorithm within interactive settings.

For the existing state-of-the-art traffic simulation algorithms, information about the computational complexity is very hard to find in the literature. Thus, the authors of system MATSim admit that finding simple predictive rules for the computational performance of their system is difficult and report only computational times for specific scenarios [11]. We could find out that the estimated complexity of agent-based simulation, as in SUMO [12], is linear with regard to the number of agents [56], which is the same as in our algorithm. However, for the sake of faster computations, the simulators use simplified network models. Thus, in MATSim, all vehicles are travelling with free-flow speed. SUMO allows users to set the maximal possible speeds for network links but specific speed-flow dependencies cannot be given. It is possible to introduce traffic lights and their operation programs, but this possibility is irrelevant to spatially abstracted transportation networks. While our algorithm simulates movements of individual vehicles with their specific routes, like the existing agent-based simulation algorithms, it takes into account link-specific speed-flow dependencies, which is not done in the other algorithms.

4.2.3 Traffic Forecasting Support

Before the simulation can take place, the analyst needs to define the scenario to be simulated. Formally speaking, the analyst needs to define the set of extra vehicles V, their origins and destinations, the routes they will follow, and the time when each vehicle starts moving. The process of scenario definition is supported by a wizard guiding the analyst through the required steps and providing visual feedback at each step. The analyst can investigate the results of each step and, if necessary, repeat the step after changing previously made settings. The procedure starts with loading the model sets TMF, DMSL and DMFS.

In step 1, the analyst specifies the number of extra vehicles and sets their places of origin. If the trips of the vehicles will originate from a few places, the analyst interactively selects the places on a map and specifies the number of vehicles in each place. If the number of origin places is large, the analyst selects these places using a suitable interactive filter (e.g., spatial or attribute-based [2]) and lets the system distribute the given number of extra vehicles among the places proportionally to place weights. Values of any numeric attribute may be used as

place weights, for instance, population number, average number of vehicles present in a place, or average number of trips starting from this place in a certain hour of a day. The result of vehicle distribution is shown on a map by proportional symbols located in the origin places.

In step 2, the analyst defines the set of destination places. Depending on the number of places, the analyst selects them interactively on a map or by filtering. The number of vehicles that will travel to each destination can be set manually or determined automatically proportionally to place weights, which are specified in the same way as for the origin places. After this step, the system displays a map with pie charts showing the assigned numbers of trip starts and ends in all places involved.

In step 3, the origins need to be matched with the destinations, i.e., for each origin, the destinations of all vehicles assigned to it need to be chosen. A recently proposed radiation model [58] defines the probability of moving between two places as a function of the local population densities. If the weights of the origin and destination places have been set based on their population densities, the radiation model is implemented by randomly choosing for each vehicle a destination place in such a way that the probability of choosing a place is proportional to its weight. If population data are not available, the number of trip ends can be used as a proxy of population density. Moreover, depending on the planned scenario, the analyst can use the number of trip ends at different times of the day. To forecast traffic to industrial and business areas, the number of trip ends in the morning hours of the weekdays can be used as a proxy of the business time population density. To forecast traffic to residential areas, the number of trip ends in the evenings can be used as a proxy of the resident population density.

In step 4, the routes for the vehicles are computed. Our system includes an implementation of Dijkstra's shortest (fastest) path algorithm [22], which uses the average durations of the moves through the links and can take into account link weights. Taking the average or maximal traffic intensities as the link weights will give preference to more "popular" links going along motorways and major roads. For each origin-destination pair, several fastest paths can be computed. A path for a vehicle is randomly chosen from available options with the probability inversely proportional to the estimated travel time.

For route generation, the analyst is expected to set several parameters: the number of alternative routes to be generated, whether to use link weights and, if so, what attribute defines the weights, and what transformation to apply to the weights (logarithmic or power). To find out what settings lead to most realistic routes, the analyst can interactively test the route generation with different settings. For this purpose, the analyst selects a pair of places in the map display and lets the system generate a specified number of routes between these places. The system represents the generated routes on the map by lines. The routes resulting from different runs of the route generation testing procedure are shown in different colors. Besides, the system creates a table with the parameters of each run and statistics of the results, including the number of generated routes and their minimal, maximal and average travel times and lengths. This allows the analyst to find appropriate settings. After the settings are made, the system generates routes for all vehicles. The results are shown on the map display in two ways: detailed (each route is shown by a line) and summarized (the routes are aggregated into overall link loads, which are represented by proportional widths of flow symbols). The paths can also be seen in a space-time cube. The user needs to keep in mind that the times in the paths are based on the average travel times between the places. More realistic times taking into account the traffic conditions can only be determined in the course of traffic simulation.

In step 5, the analyst specifies the time interval in which the vehicles will start their movement and chooses the distribution mode of the trip starts within the selected interval. It may be a uniform or normal distribution or a skewed distribution with the maximal frequency attained at the user-chosen position within the interval (in particular, it may be one of the interval ends). Optionally, the analyst may limit the temporal extent of the simulation, i.e., for how many hours the simulation will be done. The analyst also chooses the length of the time intervals by which the simulation results will be aggregated.

After this, the system runs the simulation. The results can be shown in several ways. The movements of individual vehicles can be played on an animated map, and the trajectories of the vehicles can be shown in a space-time cube. An animated map can also show the result as aggregated flows by time intervals. However, animated displays do not effectively support the assessment of the forecast. More useful are interactive time graphs linked to a map display through brushing. The time graphs show the evolution of several dynamic properties of the links: base load, extra load, total load, possible speed, number of vehicles that could move, number of suspended vehicles, etc. The simulation procedure also computes dynamic characteristics of the places, which can also be represented on time graphs. The use of the time graphs will be demonstrated by example in the next section.

4.2.4 Traffic Forecasting Example

As an example, we shall simulate the following scenario. Many people have come with their cars to the sea coast on the west of Tuscany to spend a summer weekend at the sea. At about the midday of Saturday it is

announced that a severe storm is coming from the west and is expected to hit the coast around 18 o'clock. The weekend tourists are recommended to return to their homes. They should leave the coastal area before the storm comes. Many tourists decide to follow this recommendation. As a result, about 50,000 cars generate extra traffic in addition to the regular traffic on Saturday afternoon and evening.

In trying to simulate this scenario, we need to take into account the fact that our example data do not reflect the movements of all vehicles on the study territory but only the movements of 42,686 private cars; hence, the computed traffic intensities need to be scaled, as explained in section 2. From the data provider, we know that they track about 2% of the circulating private cars in Italy (the data are used for car insurance); however, the proportion of these cars in the whole traffic is unknown. A study has been conducted to find out whether these tracks can be used to infer the real traffic counts [54]. For a small set of road sections in Pisa, the researchers compared the counts of the tracked vehicles by time intervals with real traffic counts obtained from traffic sensors. The conclusion was that the sample of tracked vehicles is highly significant and can give a good approximation of the real traffic after appropriate scaling; however, the scaling functions differ for different road segments. Since real traffic counts for the entire underlying territory of our dataset are not available, valid scaling functions for all links cannot be derived for our specific case study. However, this does not diminish the general value of our approach. It is obviously applicable when data refer to the entire population of vehicles or when scaling functions can be derived using additional data. We believe that the spread of traffic sensors will increase over time, and obtaining complete traffic counts will be less and less problematic.

For the current case, we apply the following approach. We assume that the movements of the 2% of the private cars for which we have data are representative of the whole population of the private cars. We want to forecast movement of 50,000 private cars in total, and we know that about 2% of these cars are present in the available dataset and are accounted for in our models. Based on this reasoning, we shall simulate the movement of 2% of the 50,000 private cars, i.e., 1,000 cars, and treat it as representative for the whole set of 50,000 cars.

In steps 1 and 2 of the scenario definition procedure, we need to specify the trip origins and destinations of the 1,000 cars. We have no data about the homes and weekend accommodations of real tourists. Earlier it was found that the spatial distribution of the most probable home locations of car drivers that could be extracted from the GPS tracks of their cars is very highly linearly correlated with the distribution of the resident population [28]. The most probable home locations have been determined as the most frequent locations of trip ends in the evening and night hours. Hence, the distribution of the trip ends in the evening and night can serve as a proxy for the resident population distribution, i.e., the counts of the evening trip ends in the places can be taken as the place weights for generating the spatial distribution of the possible trip destinations of the weekend tourists. Moreover, for the places located at the coast, these weights can be used for generating the spatial distribution of the supposed trip origins, assuming the number of tourists in a place to be proportional to the number of the place residents. This is a reasonable assumption since tourists usually stay in populated places.

By interacting with the map display, we divide the set of places into coastal and inland. Then we let the system automatically distribute 1,000 cars over the coastal places proportionally to the place weights, for which we take the average numbers of trip ends after 18:00. Analogously, we distribute the destinations of the car trips over the inland places. The results are shown on a map in Fig. 11 by violet and orange circles with the sizes proportional to the number of trips originating from the places (violet) and the number of trips ending in the places (orange). In our scenario, we prohibit the origin places to be also used as destinations, but, in general, this is not required.

In step 3, the system automatically matches the origins with the destinations as described previously. In step 4, we find suitable parameter settings for the vehicle routing by generating test routes for various origin-destination pairs and parameter values. After this, the system generates possible routes and chooses a route for each car. In Fig. 11, the result of the route generation and assignment is shown on the map in an aggregated form: for each link, the number of cars that will pass through it (link load) is shown by proportional width of a curved line. A small map fragment in the lower right corner demonstrates this variant of link depiction in more detail.

[Fig. 11. The distribution of the trip origins and destinations and the expected link loads.]

In step 5, we specify the time interval in which the cars will start moving: from 13:00 to 16:00 on the 7th of June 2014 (Saturday). We set that the trip starts will be normally distributed within this interval and that the simulation results need to be aggregated by 10-minute intervals. Then we start the simulation, which runs for 19 seconds, and then assess the simulation results. We see that the total time that will pass from the first vehicle starting moving (13:03) till the last vehicle arriving at its destination (23:15) will be 10 hours 10 minutes.

The simulation tool has generated time series of dynamic properties of the places and links by 10-minute intervals, as we requested. We use time graphs of these time series to investigate the progress of the evacuation. Fig. 12 demonstrates a set of time graphs showing characteristics of the places: total load (A), number of extra vehicles that arrived (B), number of extra vehicles that moved out (C), and number of suspended extra vehicles

(D). The dynamics of these characteristics in the places are represented by lines colored in violet for the places located at the coast and in orange for the remaining places. We see that the last extra vehicles will leave the coastal places by 19:30, i.e., in 1.5 hours after the supposed storm start.

[Fig. 12. Simulation results aggregated by the places and 10-minutes intervals: total load (A), number of extra vehicles that arrived (B), number of extra vehicles that moved out (C), number of suspended extra vehicles (D). The lines corresponding to the coastal places are shown in violet and the remaining lines in orange.]

The time graph D shows us that in some places many cars will be suspended for long time. The lines corresponding to these places are highlighted in black. Two places are located at the coast and two are inland. The problematic places are highlighted in black and labeled by numbers on a map in Fig. 13, where the circles represent the maximal counts of suspended cars per time interval. The areas around the problematic places are shown in more detail at the bottom of Fig. 13. The maximal suspensions on the links are represented by the widths of the curved lines. The major suspensions will occur at motorway entrances near highly populated areas (1, 4) and on motorway junctions (2, 3).

[Fig. 13. The map shows the places (red circles) and links (blue curved lines) where the evacuating cars will be queuing.]

4.3 Develop Options

Since it will take too long for the evacuating cars to leave the coastal area, it is necessary to find possible interventions that can reduce the evacuation time. Two types of interventions are possible: (1) direct a part of the traffic to alternative roads that are expected to be less busy, and (2) on motorways, use one or more lanes going in the opposite direction as additional lanes for the evacuating cars. These actions can be introduced into a scenario in the following way. Traffic re-directing can be imitated by modifying link weights. Increasing the weights of links that are not heavily used will attract more traffic to these links. Decreasing the weights of overloaded links may divert a part of the traffic to other links, if reasonable alternative routes exist. Setting a link weight to zero imitates closing this link for traffic. Using additional lanes on motorways to speed up the movement of traffic in a particular direction is modeled by modifying the maximal numbers of vehicles that can pass through links (i.e., link capacities). For links going in the target direction, the capacities are increased, and for links going in the opposite direction, the capacities are decreased. The framework allows two ways of interactive modification of link properties: by selecting links on a map one by one and entering new values or by selecting a set of links and modifying their properties in a uniform way, e.g., by multiplying by a given factor.

After finishing a simulation run, the system allows the analyst to inspect the result as long as needed, then to modify the weights or capacities of some links, and then start a new simulation. The analyst does not need to perform again the full scenario definition procedure. The tool takes all settings from the previous run and only asks the user about the attributes specifying the altered link properties. Results of two simulation runs can be compared in two ways. First, the analyst can visualize the result of each run on an individual map and individual set of time graphs, put the maps and graphs side by side, and compare them visually using brushing and synchronous filtering to link the views. Second, the system can compute and visualize the differences between the results.

Let us demonstrate the process of developing options and the possibilities for comparing results of two simulations by example. We shall try to speed up the evacuation of the cars from the coastal area of Tuscany using the two types of regulatory actions mentioned earlier. First, we shall try redirecting traffic. We observe (Fig. 11) that the road SP24 passing Pisa and connecting the coastal motorway A12 to the highway Livorno (Leghorn) - Florence (the southern one of the two parallel major roads visible in Fig. 11) will be underused. Redirecting some traffic to SP24 can reduce the suspensions in place 3. To check this, we increase the weights of the links going along SP24 by factor 10 and re-run the simulation. In the new scenario, many cars will use SP24. This is readily observed in the change map in Fig. 14. To create this visualization, we have subtracted the expected link loads in scenario 1 from the expected link loads in scenario 2, and we have done the same for the maximal counts of suspended vehicles in the places. The differences in the link loads are shown by the colors and widths of the curved lines representing the links. The links where the differences are positive (i.e., the loads increased in scenario 2 compared to scenario 1) are shown in the blue color, which was used for the links before. The links where the differences are negative (i.e., the loads decreased) are shown in the opposite orange color. The line widths are proportional to the absolute differences. We see a massive flow in blue through Pisa along the previously underused road SP4, the load of which will increase by 323 cars. The flow through place 3 will decrease by 176 cars; moreover, the flow through place 2 and on the entire northern coast-inland motorway (A11) will decrease by 131 cars.

[Fig. 14. The impact of redirecting a part of the traffic to road SP24 passing Pisa is visualized as a change map.]

The circles represent the changes of the maximal counts of suspended cars in the places. The circle areas are proportional to the absolute differences. Positive changes (i.e., increases) are shown in semi-transparent red and negative changes (i.e., decreases) in cyan. The colors of the flow lines and circles have been chosen for making a good contrast on the map. We see that the maximal number of suspended cars in place 3 will decrease by 134 and, moreover, the maximal number of suspended cars in place 2 will also decrease by 19.

We visualize the new simulation results in time graphs, as in Fig. 12, and see that not only the number of suspended cars will decrease in places 2 and 3 but also the duration of the suspension. However, the total evacuation duration will increase by 1 hour and the time required for leaving the coastal area will increase by 10 minutes. Hence, the redirection of some traffic to SP24 solves the local problem in place 3 and alleviates the problem in place 2 but does not yield a global improvement of the evacuation time. Furthermore, the local improvement in place 3 will be annulled by increased suspensions in place labeled 5 located farther on the same road (Fig. 14).

The next action we shall try is increasing the capacities of the inland-directed links located along the major roads through partial use of the opposite-going lanes of these roads for the evacuating cars. We expect that this can speed up the removal of the evacuating cars from the coastal places. We interactively select the links on the map display (the selected links are shown in red in Fig. 15) and multiply their capacities by factor 1.5. The capacities of the opposite links are multiplied by factor 0.5. This imitates using a half of the opposite link capacities for the movement out of the coastal area. After running the simulation with the modified link capacities, we observe large improvements in the evacuation from the coastal area. Time graphs, as in Fig. 12, show us that the evacuating cars will completely leave the coastal area by 19:00, which is 30 minutes earlier than in the original scenario. We select the coastal places where some cars will still be present after 18:00 and see that in most of them there will be no evacuating cars already by 18:20 and only in two places cars will be present till 18:50. By brushing the lines of these two places on the time graph and looking at a map, we find that they are located on a motorway junction near Viareggio. In Fig. 16, the numbers of cars that will leave the coastal places only after 18:00 are shown by proportional circle sizes. There will be 61 such cars (6.1% of the total number of simulated cars) in the two places near Viareggio and maximum 4 cars in all other places.

[Fig. 15. The links where the capacities will be increased by using opposite lanes are shown in red.]

[Fig. 16. The coastal places where some evacuating cars will still be present at 18:00 and later are marked by red circles with the sizes proportional to the numbers of these cars.]

Hence, re-routing the traffic and increasing the link capacities for the inland-directed movement will significantly speed up the evacuation from the risk area, but about 6% of the cars concentrating near Viareggio will not be able to leave the area before 18:00. We do not see any further opportunities for helping these cars to move out faster. It may be too dangerous for them to keep moving during the storm. A safer option may be to direct the cars suspended at the problematic junction to the nearest town Viareggio and provide the evacuees with shelters for staying until the storm ends.

We also investigate the movement of the evacuating cars beyond the coastal area. The time that will pass till the last car arrives at its destination will decrease by only 10 minutes. There will be three inland places where many cars will be suspended during long time intervals. One of them is place 2 near Lucca known from Figs. 13 and 14 and two other are located farther to the east. The places of suspensions are well visible in a space-time cube (Fig. 17) representing the simulated trajectories of the cars (the trajectories are colored according to their places of origin). The suspensions appear in the cube as vertical trajectory segments, which mean that the spatial positions do not change as the time passes. The three problematic places are marked in the cube by arrows and labeled 2, 6, and 7. Compared to the original scenario, the maximal number of suspended cars in place 2 will decrease from 115 to 80. In place 6, the maximal number of queuing cars will reach 55, and the congestion time will be quite long: from 14:30 till 21:10. Unfortunately, there are no further possibilities for improving the flow of the evacuating traffic through these places, as the opposite link capacities have been already used. Fortunately, the places are located out of the most dangerous area; hence, the congestions will not have dramatic consequences. Place 7 is located yet farther to the east. The congestion time in it (from 15:00 to 19:20) will be much shorter than in place 6.

[Fig. 17. The simulated car trajectories are represented in a space-time cube. The arrows point at the places of major suspensions.]

Hence, interactive visual tools allowed us to find suitable traffic regulation actions that will enable faster evacuation of cars from the risk area.

5 EVALUATION AND VALIDATION

Evaluation and validation of the models TMF, DMSL and DMFS take place at the 'assess' stage. The general approach is to compute and analyze the residuals between the model-predicted values and the real values. A model is good when the residuals are random, i.e., no regular patterns exist. If it is not so, the analyst refines the model, e.g., by subdividing the data subset the model refers to and building a new model for each sub-subset. The method and tools for model evaluation and validation are described in detail in an earlier paper [7].

The simulation model, which is based on the models TMF, DMSL and DMFS, also needs to be evaluated as a whole. The standard way to validate results of simulation is to compare them with real data. However, in case of predicting extraordinary traffic, this approach cannot be applied directly since real data rarely reflect unusual events that could cause extreme traffic flows.

We propose the following approach to assessing the trustability of a traffic simulation model using available historical data. The idea is to exploit the regular differences in the traffic amounts between busy and quiet hours of a day. The difference between a busy hour and a quiet hour can be taken as unusual additional traffic that suddenly appeared in the quiet hour. The progress of this additional traffic is simulated with the regular quiet hour traffic taken as the base traffic. The simulation result is compared with the regular busy hour traffic. If the traffic flows resulting from the simulation are similar to the regular traffic flows in the busy hour, the simulation model can be judged as trustable.

Let us illustrate this approach by an example. We select one work day within the time period covered by our data. Let it be Thursday, May 26, 2011. In this day, the highest traffic flows were in hour 16 (i.e., from 16:00 to 17:00). In hour 20, the traffic was much quieter. By querying the database with the original data, we obtain the counts of distinct cars that appeared on the whole territory in the time intervals 16:00-17:00 and 20:00-21:00 on May 26. These are 8,392 and 2,917 cars, respectively, the difference is 5,475. Hence, there were about 5,475 extra vehicles within the transportation network in hour 16 in comparison to hour 20. We shall simulate the movement of these additional vehicles as if they appeared in hour 20 and compare the predicted traffic flows to the real traffic flows in hour 16.

We need to generate an initial spatial distribution of the extra vehicles and specify the probable destinations of their movement. A reasonable assumption is that the additional cars are distributed proportionally to the differences in the vehicle presence counts in the places between hours 16 and 20. The computed differences are negative in only 48 (2%) out of the 2,366 places, and these negative differences range from only -6 to -1. Ignoring the few places with the small negative differences, we create an initial distribution of the 5,475 extra vehicles across the remaining places proportionally to the corresponding positive differences. The possible destinations and their capacities are defined in the same way. The trip generation is done analogously to the evacuation scenario.

We specify that the trip starts are distributed within the interval 20:00-21:00 on May 26, 2011 according to the half-normal distribution with the highest frequency of the trip starts attained at the beginning of the interval, based on the assumption that most of the extra cars were already moving at 20:00. We limit the time extent of the simulation to one hour from 20:00 to 21:00. After the simulation ends, we compare the aggregated simulation results for hour 20 with the real data for hour 16. For this purpose, we use scatterplots (Fig. 18) where the vertical axis (Y-axis) corresponds to predicted values and the horizontal axis (X-axis) to the real values. The plot on the left represents the absolute values of the traffic intensity (N of vehicles per hour). The plot on the right represents the predicted changes in the traffic intensity in hour 20 (i.e., the differences between the predicted and real values) on the Y-axis versus the real differences between the traffic intensities in hours 16 and 20.

[Fig. 18. A: The relationship between the real flows in hour 16 (X-axis) and the simulated flows for hour 20 (Y-axis). B: The relationship between the differences of the real flows in hour 16 from hour 20 (X-axis) and the differences of the simulated flows from the real flows in hour 20 (Y-axis).]

The plot in Fig. 18A shows that the estimated flows in hour 20 (Y-axis) highly correlate with the real flows in hour 16 (X-axis); the correlation coefficient is 0.836. In Fig. 18B, we see that the differences of the estimated flows to the real flows in hour 20 (Y-axis) highly correlate with the differences of the real vehicle counts in hour 16 to hour 20 (X-axis); the correlation coefficient is 0.771. We can conclude that the predicted traffic flows correspond sufficiently well to the real data. The correspondence could be improved by enhancing the routing algorithm, which currently chooses the fastest paths, leading to over-estimation the flows through the links lying on motorways and major roads and under-estimation of flows on lesser links. In reality, especially in everyday

life, people not always take fastest paths. Hence, more realistic routing would improve the accuracy of forecasts; however, developing a more sophisticated routing module is beyond the scope of this work.

The given example demonstrates the general way in which models forecasting unusual traffic can be validated. Comparisons should be done for several pairs of quiet and busy hours. In case of good correspondences between the real and predicted traffic flows, the models can be judged as sufficiently trustable.

6 DISCUSSION AND CONCLUSION

For the domain of traffic analysis and management, we have designed and implemented a visual analytics framework to support the workflow consisting of three high-level tasks: assess, forecast, and develop options. The main challenge has been to enable the exploitation of results of earlier stages at later stages of the workflow. We have achieved this by (1) creating interactive visual tools allowing the analyst to represent analysis results by formal models and (2) creating analytical tools (in particular, the simulation algorithm and its visual analytics embedding) capable of utilizing these models. The results of the 'assess' stage are models representing the temporal variation of the real traffic flows and relationships between the flow properties. At the 'forecast' stage, these models are integrated in a traffic simulation model, which produces a forecast of a traffic development scenario. The simulation model is again used at the stage 'develop options' for forecasting the effects of possible interventions in the scenario.

A distinctive feature of our approach to traffic analysis, modeling, and simulation is the use of data abstraction and generalization for modeling transportation networks and traffic properties at different levels of spatial scale. We discussed this approach with transportation researchers from the Hasselt University in Belgium (www.uhasselt.be/IMOB-EN), with whom we collaborate in a research project. They got much interested by our finding that the traffic intensities and the mean speeds in a generalized transportation network are linked by relationships (Figs. 6 and 8) analogous to the known fundamental relationships existing at the level of street segments. They appreciated the possibility of deriving the flow-speed relationships from real data and subsequent using them for traffic simulation and the possibility of simulating movements in generalized transportation networks at different spatial scales. To cross-check these possibilities, the partners are now trying to use the generalized network and the flow-speed models that we derived from the Tuscany data in a standard traffic simulator. For this purpose, the models need to be incorporated in the simulator, which is not an easy task (that is why we developed our own simulation algorithm capable of direct use of models derived from data).

We also demonstrated our framework to representatives of BBK – German Federal Department for Civil Protection and Disaster Management. They were excited by the principal possibility of rapid simulation of mass movements in emergency situations; however, they noted that civil protection agencies usually have no access to traffic data that could be used for this purpose. A way to overcome this is establishing collaboration agreements with companies collecting traffic data so that the data or, alternatively, simulation services could be provided to civil protection agencies in emergency cases.

Our research shows that visual analytics methods can help analysts not only to gain understanding (i.e., a mental model) of a phenomenon represented by data, but also to transform this mental model into explicit formal models and use these models for forecasting expectable developments and consequences of possible interventions. Our work gives a first example of how all three tasks of visual analytics, 'assess', 'forecast', and 'develop options', can be supported as a single workflow. We have generalized our experience in a general scheme (Fig. 1), which can be used in developing comprehensive visual analytics support for other applications.

REFERENCES

[1] S. Afzal, R. Maciejewski, and D.S. Ebert. Visual analytics decision support environment for epidemic modeling and response evaluation. In Proc. IEEE Conf. Visual Analytics Science and Technology (VAST'2011), pp. 191–200, 2011.

[2] G. Andrienko, N. Andrienko, P. Bak, D. Keim, and S. Wrobel. Visual Analytics of Movement. Springer, 2013.

[3] G. Andrienko, N. Andrienko, S. Rinzivillo, M. Nanni, D. Pedreschi, and F. Giannotti, Interactive Visual Clustering of Large Collections of Trajectories, In Proc. IEEE Symp. Visual Analytics Science and Technology (VAST'09), pp. 3-10, 2009.

[4] N. Andrienko and G. Andrienko, Exploratory Analysis of Spatial and Temporal Data: A Systematic Approach, Springer, Berlin, 2006.

[5] N. Andrienko and G. Andrienko, Spatial Generalization and Aggregation of Massive Movement Data, IEEE Trans. Visualization and Computer Graphics, 17(2): 205-219, 2011.

[6] N. Andrienko and G. Andrienko. Visual analytics of movement: An overview of methods, tools and procedures. Information Visualization, 12(1): 3-24, 2013.

[7] N. Andrienko and G. Andrienko, A Visual Analytics Framework for Spatio-temporal Analysis and Modeling, Data Mining and Knowledge Discovery, 27(1): 55-83, 2013.

[8] N. Andrienko, G. Andrienko, and S. Rinzivillo. Exploiting Spatial Abstraction in Predictive Analytics of Vehicle Traffic. ISPRS International Journal of Geo-Information, 4(2): 591-606, 2015.

[9] P. Bak, M. Marder, S. Harary, A. Yaeli, and H.J. Ship, Scalable Detection of Spatiotemporal Encounters in Historical Movement Data, Computer Graphics Forum, 31(3-31): 915-924, June 2012.

[10]P. Bak, E. Packer, H. Ship, and D. Dotan, Algorithmic and Visual Analysis of Spatiotemporal Stops in Movement Data. In Proceedings of the 20th ACM SIGSPATIAL International Con-ference on Advances in Geographic Information Systems (ACM GIS 2012), November 6-9, 2012. Redondo Beach, CA, USA, 2012.

[11] M. Balmer, M. Rieser, K. Meister, D. Charypar, N. Lefebvre, K. Nagel, and K.W. Axhausen. MATSim-T: Architecture and Simulation Times. In A.L.C. Bazzan and F. Klügl (eds.) Multi-Agent Systems for Traffic and Transportation Engineering, 57–78, Information Science Reference, Hershey, 2009.

[12]M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, SUMO - Simulation of Urban MObility: An Overview, In Proc. 3rd Int. Conf. Advances in System Simulation (SIMUL 2011), pp. 63-68, 2011.

[13]M. Bögl, W. Aigner, P. Filzmoser, T. Lammarsch, S. Miksch, and A. Rind, Visual Analytics for Model Selection in Time Series Analysis, IEEE Trans. Visualization and Computer Graphics, 19(12): 2237-2246, 2013.

[14]E. Bourrel and J.-B. Lesort, Mixing Micro and Macro Representations of Traffic Flow: A Hybrid Model Based on the LWR Theory, 82th Ann. Meeting of the Transportation Research Board, Washington DC, 2003.

[15]W. Burghout, H.N. Koutsopoulos, and I. Andreasson, Hybrid Mesoscopic-Microscopic Traffic Simulation, Transportation Research Record, 1034: 218-225, 2005.

[16]W. Burghout, H.N. Koutsopoulos, and I. Andreasson, A Discrete-Event Mesoscopic Traffic Simulation Model for Hybrid Traffic Simulation, Proc. IEEE Intelligent Transportation Systems Conf. (ITSC'06), pp. 1102-1107, 2006.

[17]T. Crnovrsanin, C. Muelder, C. Correa, and K-L. Ma, Proximity-based Visualization of Movement Trace Data, In: Proceedings of the IEEE Symposium on Visual Analytics Science and Technology (VAST) 2009; IEEE Computer Society Press, pp. 11-18, 2009.

[18]F.C. Daganzo, The Cell Transmission Model: A Dynamic Representation of Highway Traffic Consistent with the Hydrodynamic Theory, Transportation Research Part B: Methodological, 28(4): 269-287, 1994.

[19]F.C. Daganzo, The Cell Transmission Model Part II: Network Traffic, Transportation Research Part B: Methodological, 29(2): 79-93, 1995.

[20]J.M. DelCastillo and F.G. Benitez, On the Functional Form of the Speed-Density Relationship I: General Theory, Transportation Research Part B: Methodological, 29(5): 373-389, 1995.

[21]U. Demšar, A.S. Fotheringham, and M. Charlton. Exploring the spatio-temporal dynamics of geographical processes with Geographically Weighted Regression and Geovisual Analytics. Information Visualization, 7: 181-197, 2008.

[22]E.W. Dijkstra, A Note on Two Problems in Connexion with Graphs, Numerische Mathematik, 1: 269–271, 1959.

[23]J.A. Dykes and D.M. Mountain, Seeking structure in records of spatio-temporal behaviour: visualization issues, efforts and applications. Computational Statistics & Data Analysis; 43:581-603, 2003.

[24]S. van den Elzen and J.J. van Wijk, BaobabView: Interactive construction and analysis of decision trees, In Proc. IEEE Conf. Visual Analytics Science and Technology (VAST'11), pp. 151-160, 2011.

[25]S. Garg, J.E. Nam, I.V. Ramakrishnan, and K. Mueller, Model-driven Visual Analytics, In Proc. IEEE Symp. Visual Analytics Science and Technology (VAST'08), pp.19-26, 2008.

[26]S. Garg, S., I.V. Ramakrishnan, and K. Mueller, A Visual Analytics Approach to Model Learning, In Proc. IEEE Symp. Visual Analytics Science and Technology (VAST'10), pp.67-74, 2010.

[27]D.C. Gazis, Traffic Theory, Kliwer Academic, Boston, USA, 2002.

[28]F. Giannotti, M. Nanni, D. Pedreschi, F. Pinelli, C. Renso, S. Rinzivillo, and R. Trasarti, Unveiling the complexity of human mobility by querying and mining massive trajectory data, VLDB Journal, 20(5): 695-719, 2011.

[29]F. Giannotti and D. Pedreschi, editors. Mobility, Data Mining and Privacy – Geographic Knowledge Discovery. Springer, 2008.

[30]M. Gleicher. Explainers: Expert Explorations with Crafted Projections, IEEE Trans. Visualization and Computer Graphics, 19(12): 2042-2051, 2013.

[31]J. Gudmundsson, P. Laube, and T. Wolle. Computational Movement Analysis. In W. Kresse and D. M. Danko, editors, Springer Handbook of Geographic Information, pages 423–438. Springer Berlin Heidelberg, 2012.

[32]H. Guo, Z. Wang, B. Yu, H. Zhao, and X. Yuan, TripVista: Triple Perspective Visual Trajectory Analytics and its application on microscopic traffic data at a road intersection. In: Proceedings of the Pacific Visualization Symposium PacificVis 2011, IEEE, pp. 163-170, 2011.

[33]Z. Guo, M.O. Ward, and E.A. Rundensteiner, Model Space Visualization for Multivariate Linear Trend Discovery, Proc. IEEE Symp. Visual Analytics Science and Technology (VAST'09), pp.75-82, 2009.

[34]M.C. Hao, H. Janetzko, S. Mittelstädt, W. Hill, U. Dayal, D.A. Keim, M. Marwah, and R.K. Sharma, A Visual Analytics Approach for Peak-Preserving Prediction of Large Seasonal Time Series, Computer Graphics Forum, 30(3): 691-700, 2011.

[35]D. Helbing, Derivation of a Fundamental Diagram for Urban Traffic Flow, The European Physical Journal B, 70: 229-241, 2009.

[36]C.C. Holt. Forecasting seasonals and trends by exponentially weighted moving averages. International Journal of Forecasting, 20(1): 5-10, January-March 2004.

[37]C. Hurter, B. Tissoires, and S. Conversy, FromDaDy: Spreading aircraft trajectories across views to support iterative queries. IEEE Transactions on Visualization and Computer Graphics, 15(6): 1017-1024, 2009.

[38]S.L.J. Jones, A.J. Sullivan, N. Cheekoti, M.D. Anderson, D. Malave, Traffic Simulation Software Comparison Study, UTCA Report 02217, University Transportation Center of Alabama, University of Alabama, USA, 2004.

[39]T. Kapler and W. Wright, GeoTime information visualization. Information Visualization, 4(2): 136-146, 2005.

[40]D.A. Keim, J. Kohlhammer, G. Ellis, and F. Mansmann. Mastering the information age-solving problems with visual analytics. Eurographics, 2010.

[41]A. Konev, J. Waser, B. Sadransky, D. Cornel, R.A.P. Perdigão, Z. Horváth, and M.E. Gröller. RunWatchers: Automatic Simulation-Based Decision Support in Flood Management. IEEE Trans. Visualization and Computer Graphics, 20(12): 1873-1882, 2014.

[42]G. Kotushevski and K.A. Hawick, A Review of Traffic Simulation Software, Computational Science Technical Note CSTN-095, Computer Science, Massey University, Auckland, New Zealand, 2009.

[43]M.-J. Kraak and F. Ormeling, Cartography: visualization of spatial data. Second edition, Pearson Education Ltd, Harlow, UK, 2003.

[44]M.H. Lighthill and G.B. Whitham, On Kinematic Waves II: A Theory of Traffic Flow on Long Crowded Roads, Proc. Royal Society of London A229, 1178 (May): 317-345, 1955.

[45]P. Lundblad, O. Eurenius, and T. Heldring, Interactive Visualization of Weather and Ship Data. In: Proceedings of the 13th International Conference on Information Visualization IV2009. IEEE Computer Society Press, pp. 379-386, 2009.

[46]R. Maciejewski, P. Livengood, S. Rudolph, T.F. Collins, D.S. Ebert, R.T. Brigantic, C.D. Corley, G.A. Muller, and S.W. Sanders, A Pandemic Influenza Modeling and Visualization Tool, Journal of Visual Languages and Computing, 22: 268-278, 2011.

[47]K. Matković, D. Gračanin, M. Jelović, A. Ammer, A. Lež, and H. Hauser, Interactive Visual Analysis of Multiple Simulation Runs Using the Simulation Model View: Understanding and Tuning of an Electronic Unit Injector, IEEE Trans. Visualization and Computer Graphics, 16(6): 1449-1457, 2010.

[48]K. Matković, D. Gračanin, M. Jelović, and Y. Cao, Adaptive Interactive Multi-Resolution Computational Steering for Complex Engineering Systems. In Proc. EuroVA 2011, Bergen, Norway, pp. 45-48, 2011.

[49]M. Migut and M. Worring. Visual Exploration of Classification Models for Risk Assessment. In Proc. IEEE Symp. Visual Analytics Science and Technology VAST'10, pp. 11-18, 2010.

[50]A. Monreale, G. Andrienko, N. Andrienko, F. Giannotti, D. Pedreschi, S. Rinzivillo, and S. Wrobel. Movement Data Anonymity through Generalization. Transactions on Data Privacy, v.3 (3): 91-121, 2010

[51]T. Mühlbacher and H. Piringer. A Partition-Based Framework for Building and Validating Regression Models, IEEE Trans. Visualization and Computer Graphics, 19(12): 1962-1971, 2013.

[52]K. Nagel and M. Schreckenberg, A Cellular Automaton Model for Freeway Traffic, Journal de Physique I, 2(12): 2221-2229, 1992.

[53]G. Newell, G, Nonlinear Effects in the Dynamics of Car Following, Operations Research, 9(2): 209-229, 1961.

[54]L. Pappalardo, S. Rinzivillo, Z. Qu, D. Pedreschi, and F. Giannotti, Understanding the Patterns of Car Travel, European Physics Journal Special Topics, 215:61–73, 2013.

[55]H. Ribicic, J. Waser, R. Fuchs, G. Blöschl, E. Gröller, Visual Analysis and Steering of Flooding Simulations, IEEE Trans. Visualization and Computer Graphics, 19(6): 1062-1075, 2013.

[56]J. Sewall, D. Wilkie, P. Merrell, and M.C. Lin, Continuum Traffic Simulation, Computer Graphics Forum, 29(2): 439-448, 2010.

[57]J. Sewall, D. Wilkie, and M.C. Lin, Interactive Hybrid Simulation of Large-Scale Traffic, ACM Transactions on Graphics, 30(6), Article 135, 2011.

[58]F. Simini, M.C. Gonzalez, A. Maritan, and A.-L. Barabasi, A Universal Model for Mobility and Migration Patterns, Nature, 484(7392): 96-100, 2012.

[59]T.A. Slocum, R.B. McMaster, F.C. Kessler, and H.H. Howard, Thematic Cartography and Geovisualization. Third Edition, Pearson Prentice Hall, Upper Saddle River, NJ, 2009.

[60]D. Spretke, H. Janetzko, F. Mansmann, P. Bak, B. Kranstauber, and M. Mueller, Exploration through Enrichment: A Visual Analytics Approach for Animal Movement. In Proceedings of 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL GIS 2011), 421-424, 2011.

[61]J.J. Thomas and K.A. Cook, eds. Illuminating the Path. IEEE Computer Society, Los Alamitos, California, USA, 2005.

[62]W. Tobler, Experiments in Migration Mapping by Computer, The American Cartographer, 14(2): 155-163, 1987.

[63]C. Ware, R. Arsenault, M. Plumlee, and D. Wiley, Visualizing the Underwater Behaviour of Humpback Whales. IEEE Computer Graphics and Applications, 26(4): 14-18, 2006.

[64]J. Waser, R. Fuchs, H. Ribicic, B. Schindler, G. Blöschl, E. Gröller, World Lines, IEEE Trans. Visualization and Computer Graphics, 16(6): 1458-1467, 2010.

[65]N. Willems N, H. van de Wetering, and J.J. van Wijk, Visualization of vessel movements. Computer Graphics Forum (CGF) 28(3): 959-966, 2009.

[66]J. Wood, J. Dykes, and A. Slingsby, Visualisation of Origins, Destinations and Flows with OD Maps. The Cartographic Journal, 47(2): 117 – 129, 2010.

[67]J. Wood, A. Slingsby, and J. Dykes. Visualizing the dynamics of London's bicycle hire scheme. Cartographica, 46(4): 239-251, 2011.

[68]M. Wörner and T. Ertl, Visual Analysis of Public Transport Vehicle Movement. In: Proceedings of International Workshop on Visual Analytics (EuroVA 2012), pp. 79-83, 2012.

[69]L. Xiao, J. Gerth, and P. Hanrahan, Enhancing Visual Analysis of Network Traffic Using a Knowledge Representation, In Proc. IEEE Symp. Visual Analytics Science and Technology (VAST'06), pp. 107-114, 2006.

List of Figures

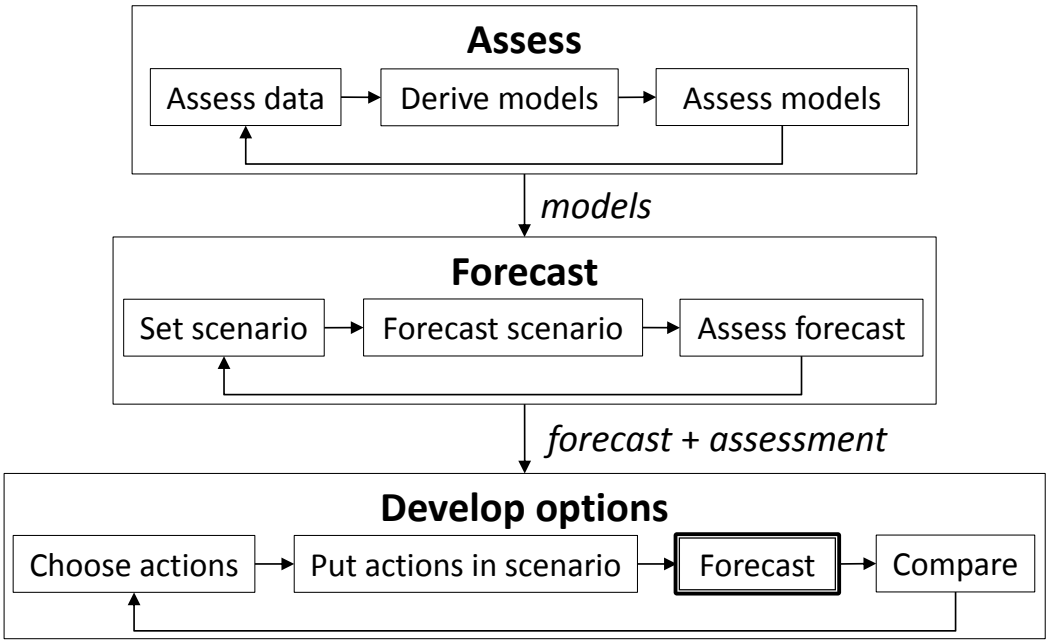

Fig. 1. Three types of visual analytics tasks as stages of a single analytical process.

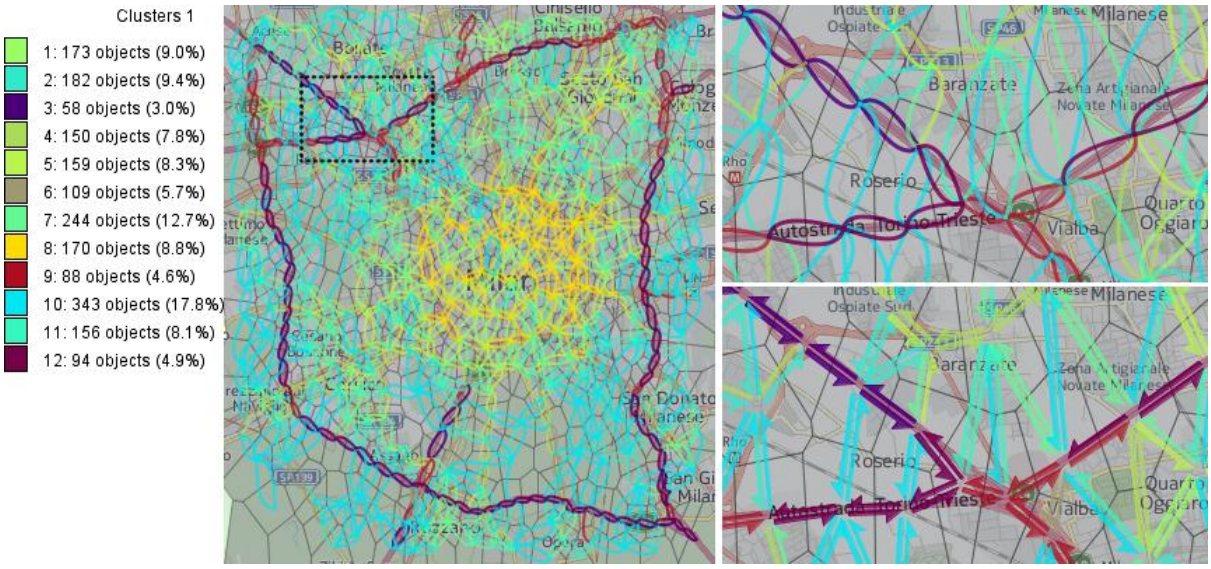

Fig. 2. A spatially abstracted transportation network of Milan (Italy) with cell radii ≈ 1km.

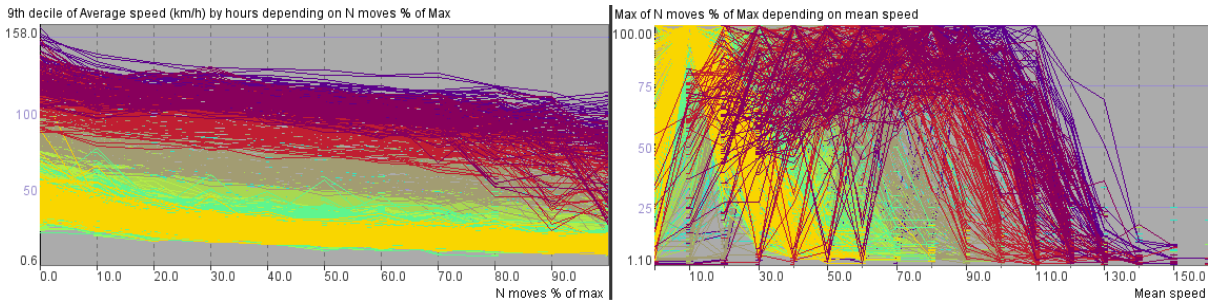

Fig. 3. The graphs represent the interdependencies between the traffic intensity and mean speed for the link of the abstracted transportation network of Milan shown in Fig. 2.

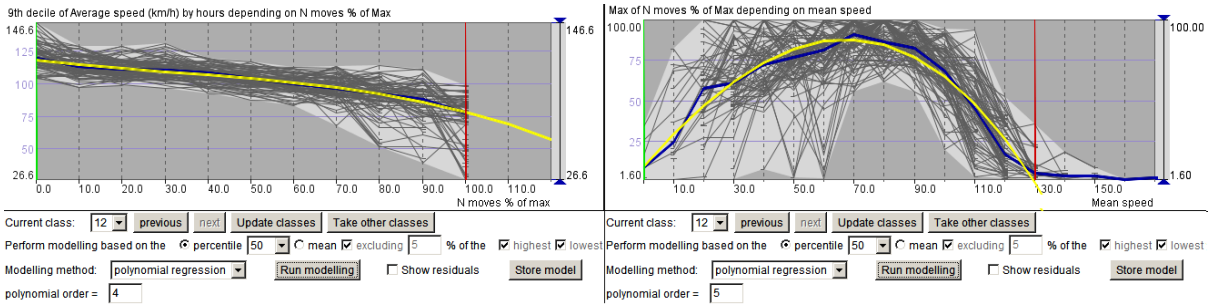

Fig. 4. The dependencies between the traffic intensity and mean speed can be represented by polynomial regression models.

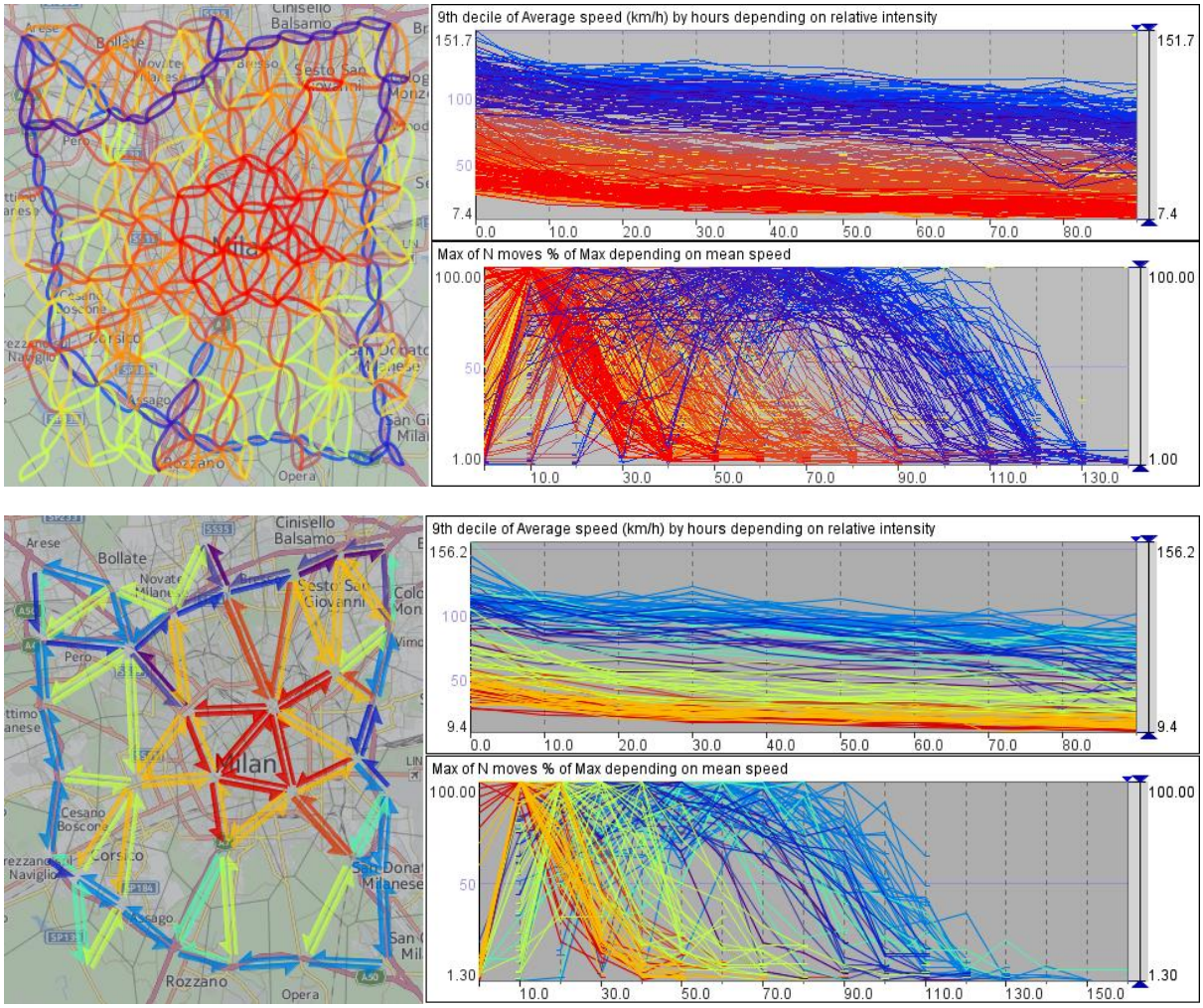

Fig. 5. The maps show spatially abstracted transportation networks of Milan with cell radii ≈ 2km (top) and 4 km (bottom). The graphs to the right of each map represent the dependencies between the relative traffic intensities and the mean speeds on the network links.

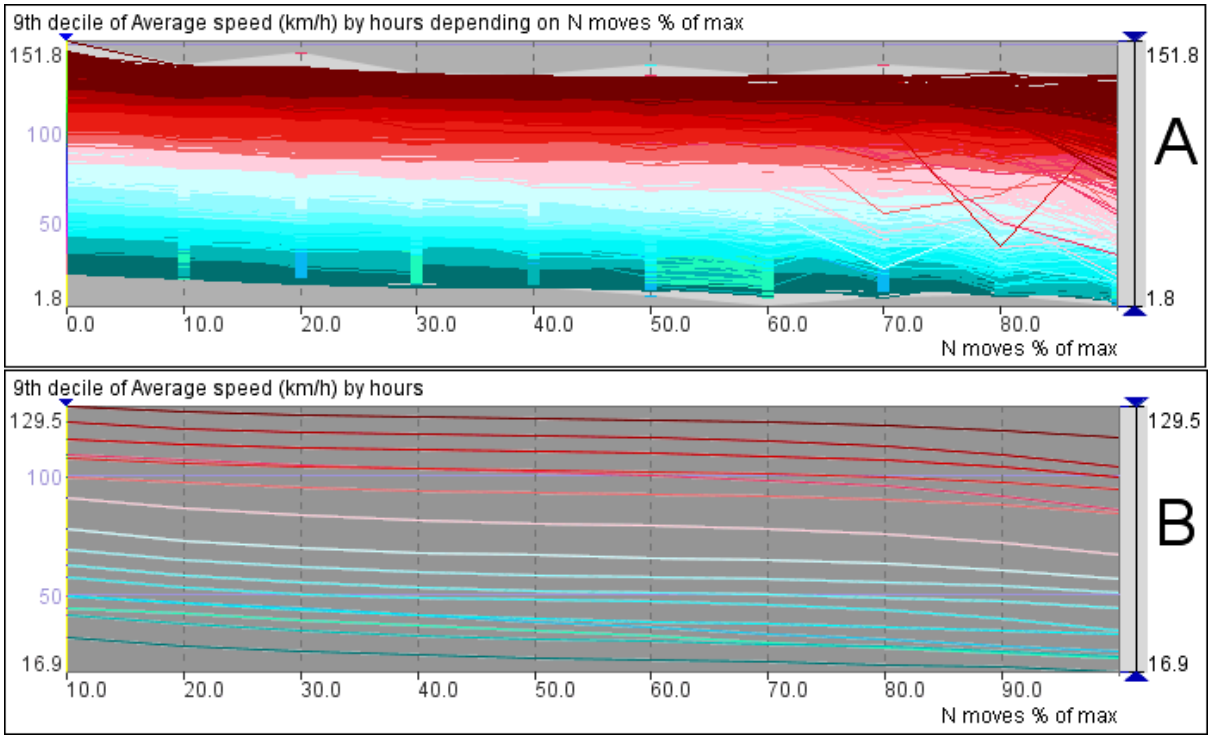

Fig. 6. A: The dependency series of the mean speed versus the traffic intensity have been clustered by similarity. B: The dependencies are represented by polynomial or linear regression models.

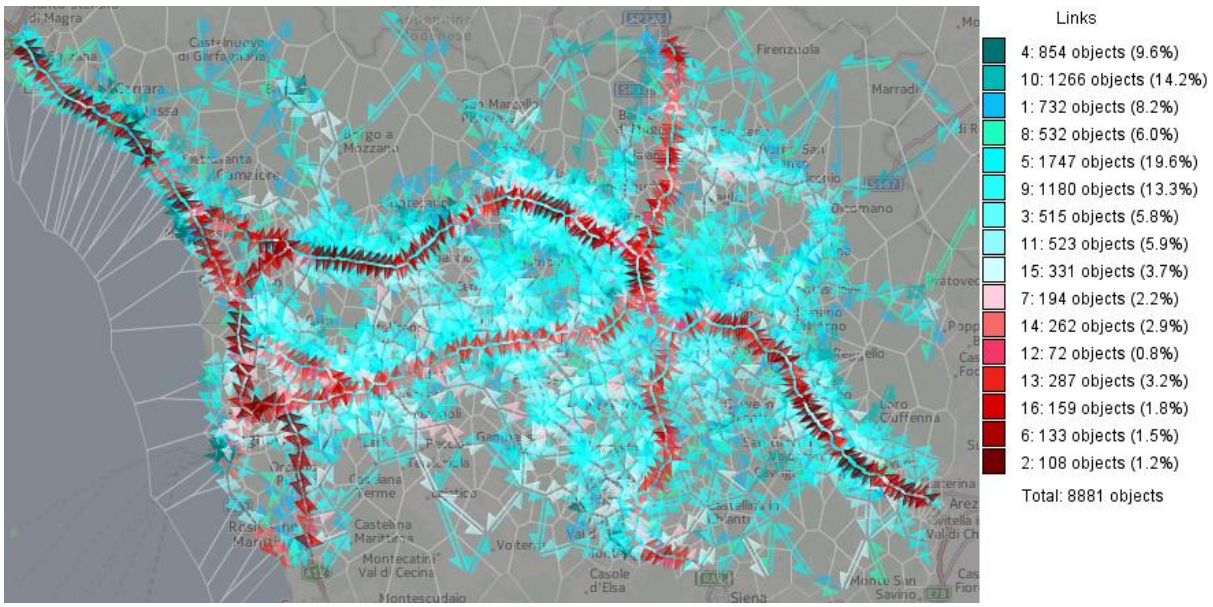

Fig. 7. The links are colored according to the cluster membership of the dependencies of the mean speeds on the traffic intensities.

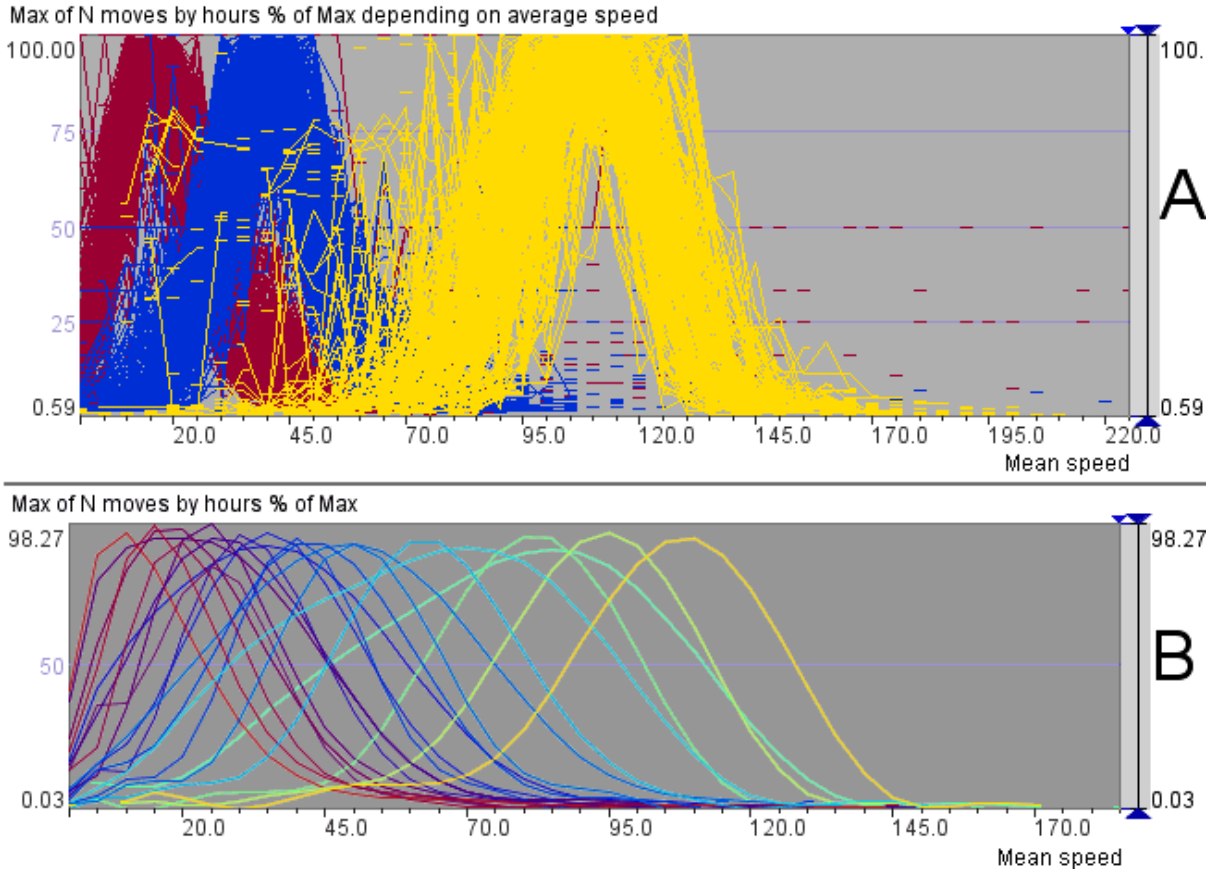

Fig. 8. A: Three selected clusters of dependency series of the traffic intensity depending on the mean speed. B: The dependency curves built for all clusters.

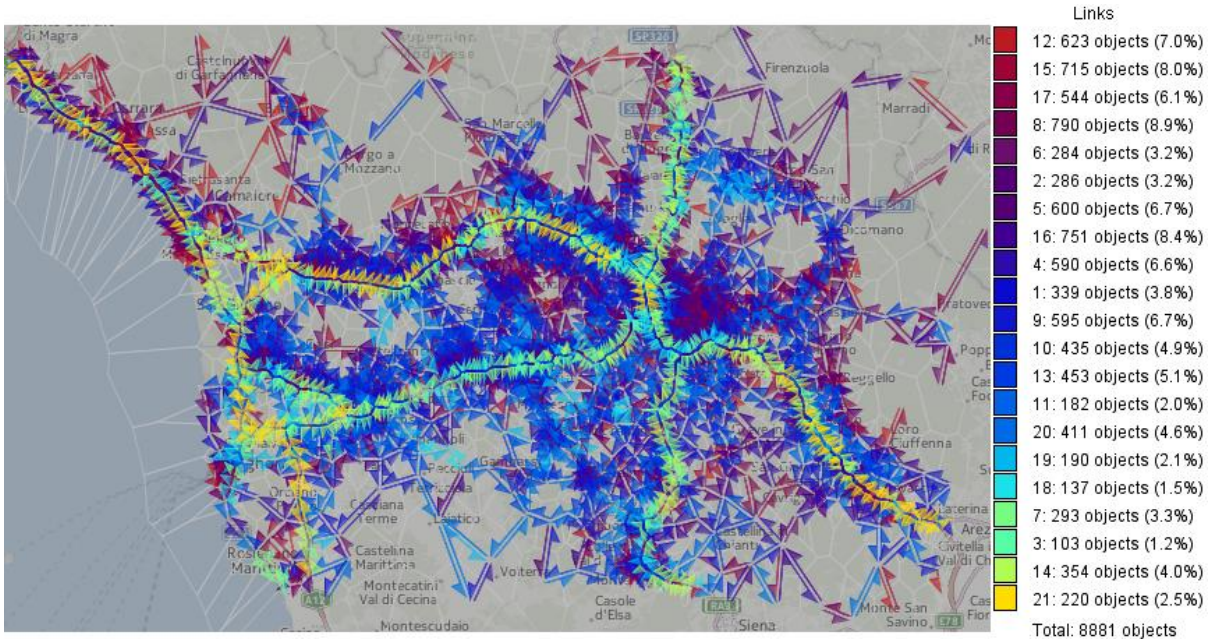

Fig. 9. The links are colored according to the cluster membership of the dependencies of the traffic intensities on the mean speeds.

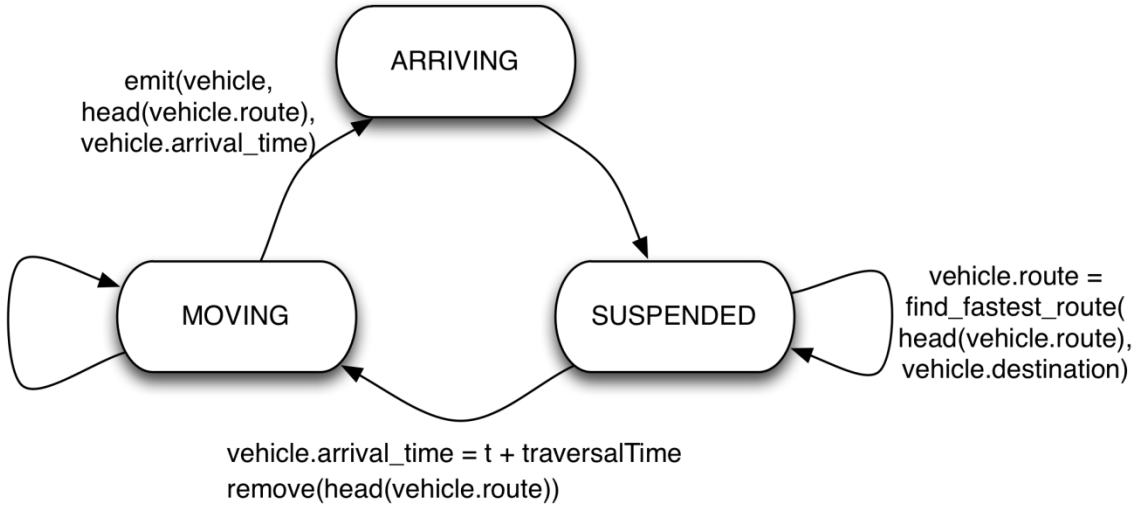

Fig. 10. The possible states of vehicles and transitions between the states in the course of the simulation.

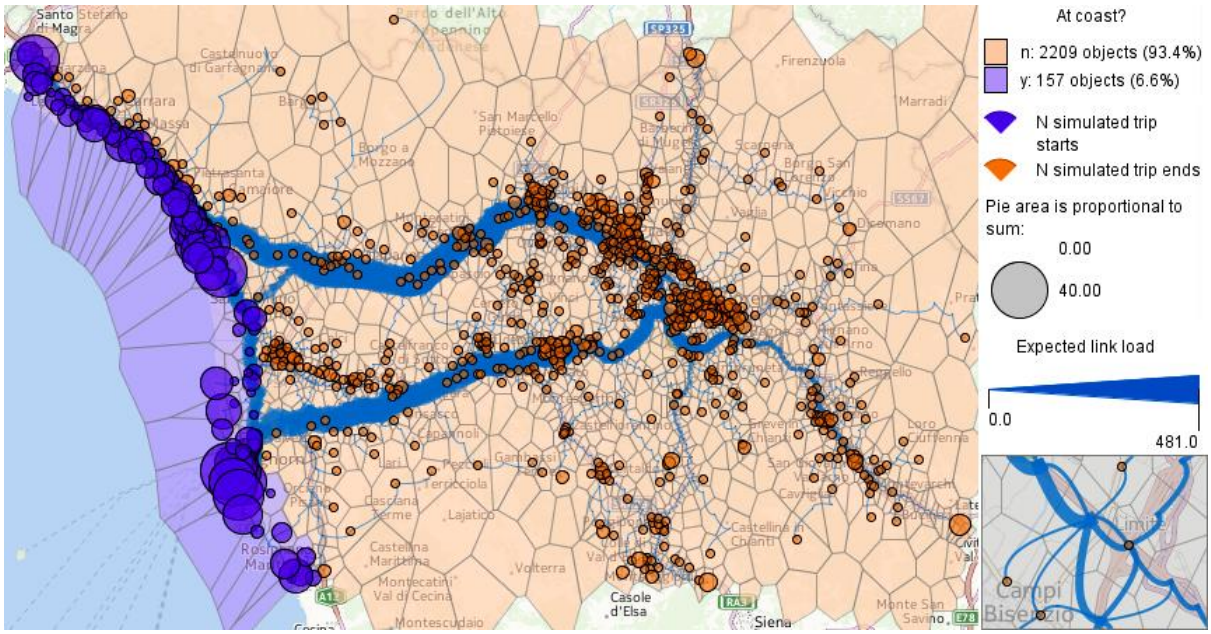

Fig. 11. The distribution of the trip origins and destinations and the expected link loads.

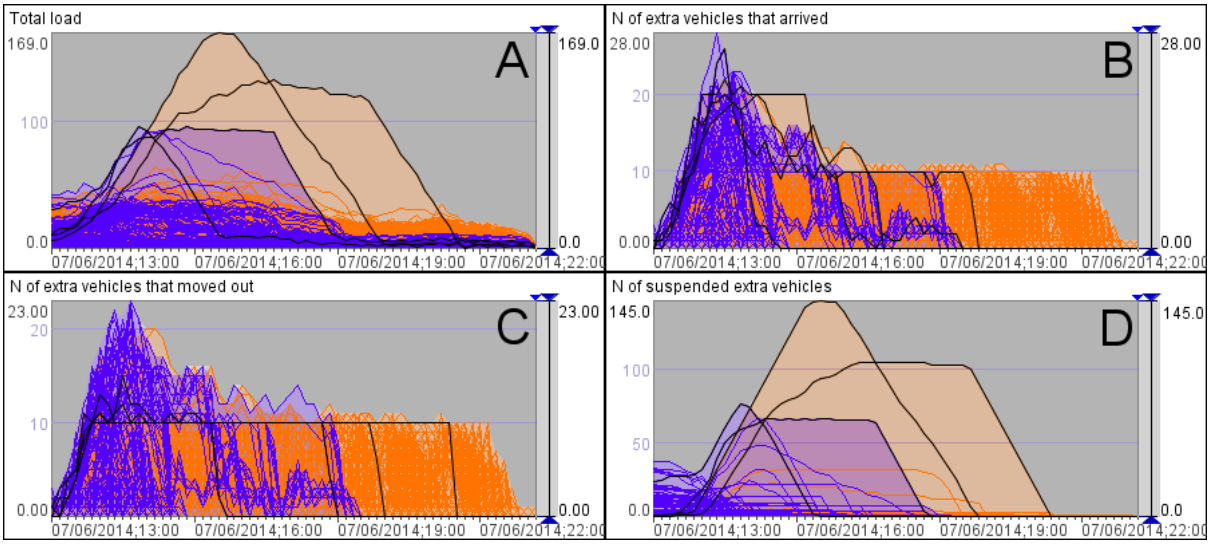

Fig. 12. Simulation results aggregated by the places and 10-minutes intervals: total load (A), number of extra vehicles that arrived (B), number of extra vehicles that moved out (C), number of suspended extra vehicles (D). The lines corresponding to the coastal places are shown in violet and the remaining lines in orange.

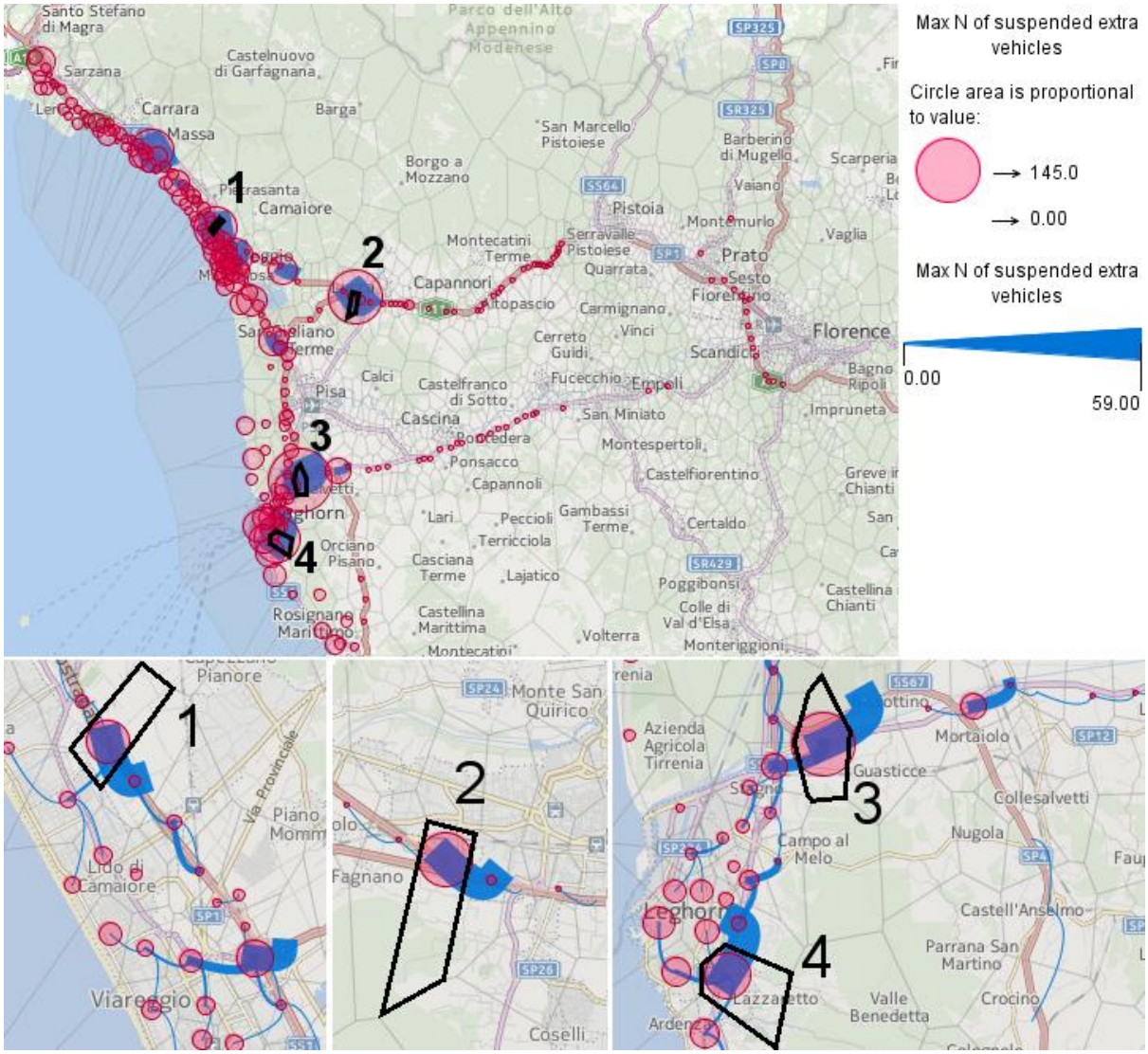

Fig. 13. The map shows the places (red circles) and links (blue curved lines) where the evacuating cars will be queuing.

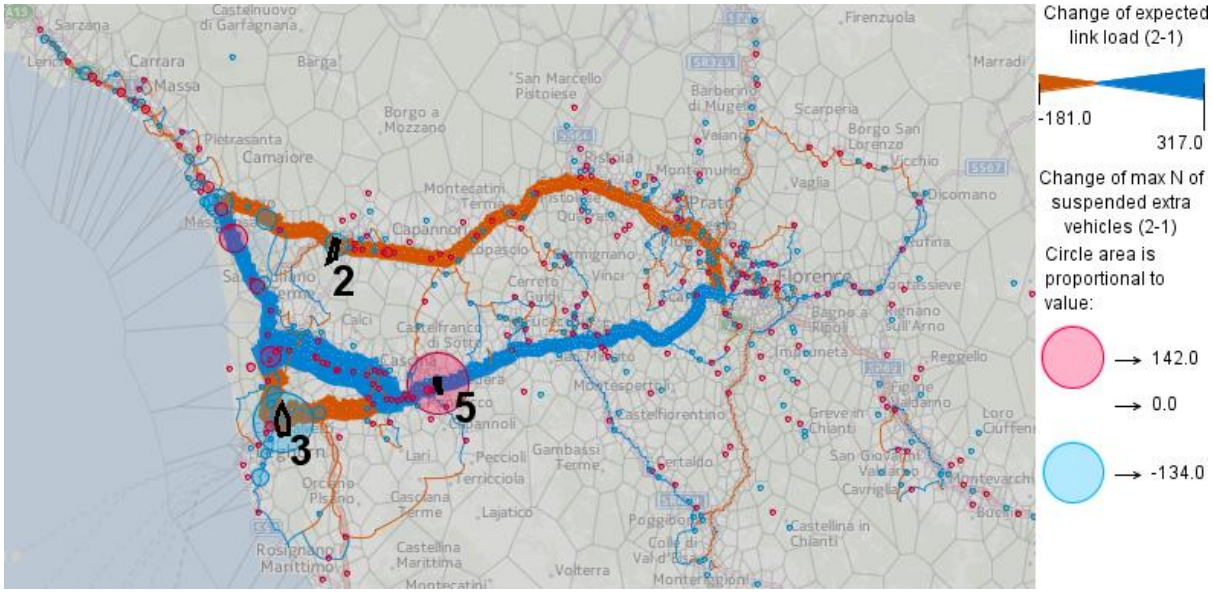

Fig. 14. The impact of redirecting a part of the traffic to road SP24 passing Pisa is visualized as a change map.

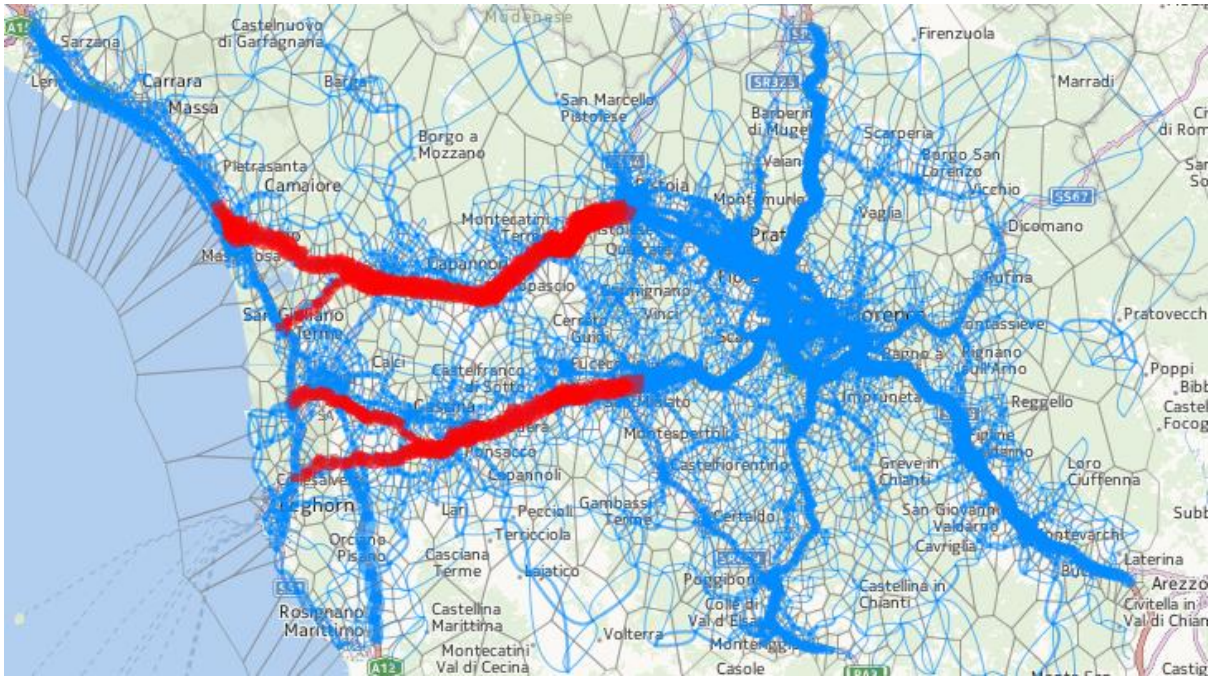

Fig. 15. The links where the capacities will be increased by using opposite lanes are shown in red.

Fig. 16. The coastal places where some evacuating cars will still be present at 18:00 and later are marked by red circles with the sizes proportional to the numbers of these cars.

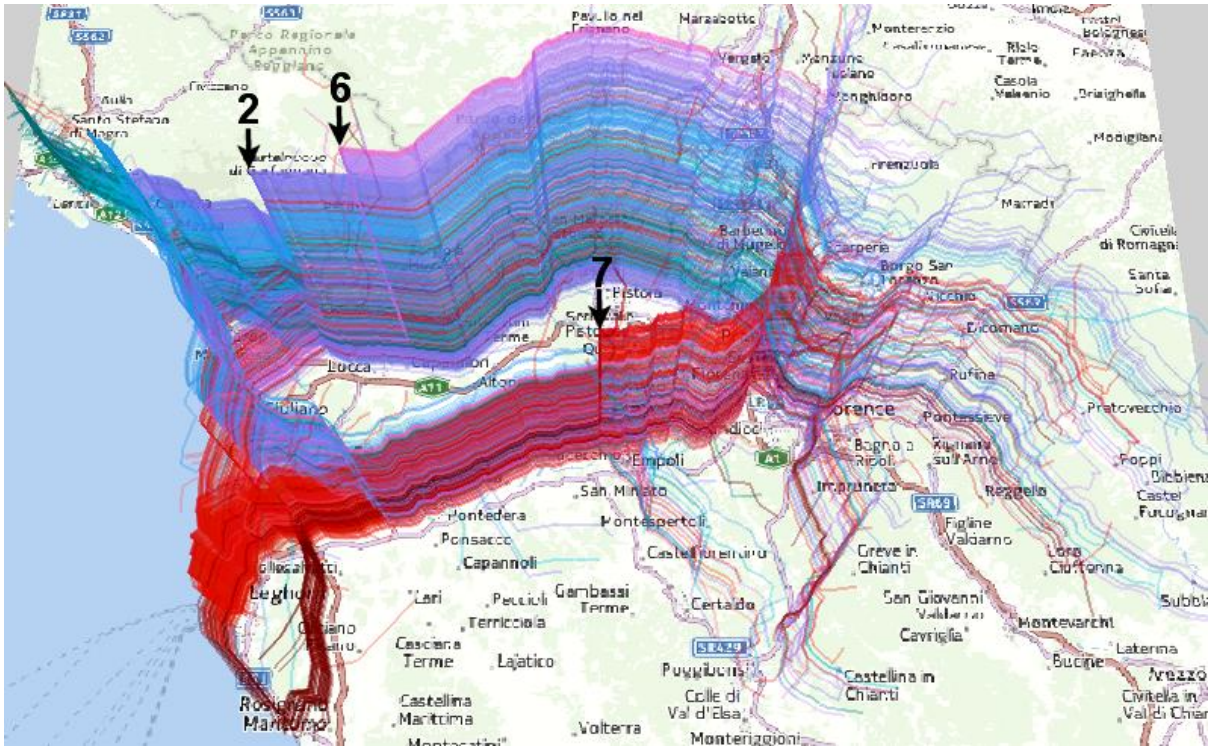

Fig. 17. The simulated car trajectories are represented in a space-time cube. The arrows point at the places of major suspensions.

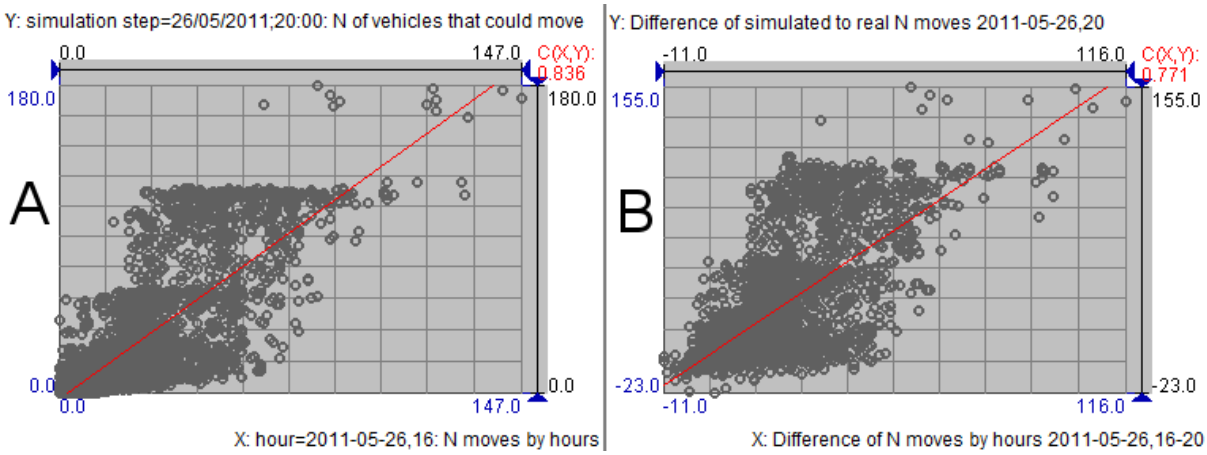

Fig. 18. A: The relationship between the real flows in hour 16 (X-axis) and the simulated flows for hour 20 (Y-axis). B: The relationship between the differences of the real flows in hour 16 from hour 20 (X-axis) and the differences of the simulated flows from the real flows in hour 20 (Y-axis).