Journal Title
XX(X):1–20

DOI: 10.1177/ToBeAssigned
www.sagepub.com/
SAGE

State Transition Graphs for Semantic Analysis of Movement Behaviours

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

Abstract

A behaviour can be defined as a sequence of states or activities occurring one after another. A behaviour consisting of a finite number of re-occurring states/activities may be represented by a directed weighted graph with nodes and edges corresponding, respectively, to the possible states and transitions between them, while the weights represent the probabilities or frequencies of the state and transition occurrences. The same applies to multiple behaviours sharing the same set of possible states. In analysis of movement data, state transition graphs can be used to represent semantic abstractions of mobility behaviours, where states correspond to semantic categories of visited places (such as 'home', 'work', 'shop', etc.), activities of moving objects ('driving', 'walking', 'exercising', etc.), or characteristics of the movement ('straight movement', 'sharp turn', 'acceleration', 'stop', etc.). Such a representation supports the exploration and analysis of the semantic aspect (i.e., the meaning or purposes) of movement. For comprehensive analysis of movement data, state transition graphs need to be combined with representations reflecting the spatial and temporal aspects of the movement. This requires appropriate coordination between different visual displays (graphs, maps, and temporal views) and appropriate reaction to analytical operations applied to any of the representations of the same data. We define in an abstract way the reactions of a graph display to analytical operations of querying, partitioning, and direct selection. We also propose visual and interactive display features supporting comparisons between data subsets and between results of different operations. We demonstrate the use of the display features by examples of real-world and synthetic data sets.

Keywords

Mobility, trajectories, transformation of movement data, coordinated multiple views, Visual Analytics

Introduction

We define a *behaviour* as a chronological sequence of states, such as performed actions, visited places, or events. A *state transition graph* (STG) is an aggregate representation of a behaviour by a directed weighted graph, where the nodes stand for the possible states and the edges for the transitions between the states. The weights of the nodes and edges may represent the frequencies or probabilities of the states and transitions, their absolute or relative durations, or other attributes. STGs are used in mathematical disciplines as visual representations of Markov chains, i.e., random processes in which the transition from one state to another depends solely on the current state and not on the sequence of states that preceded it.

Representation in the form of a state transition graph may be suitable when a behaviour consists of a finite number of re-occurring states. Very long state sequences can be effectively summarised in this way. Moreover, an STG can summarise not only a single behaviour but also an arbitrarily large set of behaviours that occurred in the same state space (i.e., when the same states re-occur in multiple behaviours). The STG representation is appropriate for analysis tasks focusing on frequencies and/or durations of individual states and transitions between them. It can be used when the exact timing of the states is not important for the analysis and there is also no need to analyse sub-sequences consisting of more than two states. These limitations of applicability of the STG representation still leave a broad range of domains and analysis tasks where this representation can be useful.

In particular, state transition graphs can be used in exploration and analysis of the semantic aspect of behaviours of moving objects, i.e., the meanings or purposes of the movements. States can represent semantic categories of visited places (such as 'home', 'work', 'shop', etc.), activities of moving objects ('driving', 'walking', 'exercising', etc.), or characteristics of the movement ('straight movement', 'sharp turn', 'acceleration', 'stop', etc.). This representation is an abstraction devoid of specific spatial and temporal information. Hence, it is not a substitute but a complement to other representations of movement data, which reflect the spatial and temporal aspects of the data. Comprehensive analysis of movement data with the use of different complementary representations requires proper coordination between tools operating on these different representations. Analytical operations applied to one of the representations need to have appropriate impacts on the other representations. For example, when a subset of trajectories is

[1]Fraunhofer Institute IAIS, Sankt Augustin, DE
[2]City University London, UK

Corresponding author:
Gennady Andrienko, Fraunhofer Institute IAIS, Schloss Birlinghoven, Sankt Augustin, 53757 Germany.
Email: gennady.andrienko@iais.fraunhofer.de

selected according to spatial criteria, state transition graphs representing semantics of the selected trajectories need to be presented to the analyst's view.

The goal of this paper is to define a general approach to using state transition graphs in movement analysis. We wish to define the approach at the level of abstract "meta-tools" and "meta-operations", that is, general categories of tools and operations. We describe the required properties of an STG display and its dynamic behaviour in response to general types of analytical operations, namely, querying, partitioning, and direct selection, regardless of any conceivable implementations of these operations. Thus, partitioning can be done using various kinds of clustering, classification, or arbitrary interactive division of a data set. Regardless of these possibilities, we consider 'partitioning' as a single meta-operation.

To facilitate understanding of this meta-level discussion, we include illustrated examples using a particular implementation of an STG tool. However, this implementation is used for merely illustrative purposes and should not be considered as our research contribution. The intended contribution of our paper consists of (1) description of data transformations used for deriving STG representations of singular and multiple individual and collective movement behaviours from trajectories of moving objects, (2) general definition of the properties and dynamic behaviour of an STG display in response to analytical operations on data, and (3) showing by examples how STG representations can be utilised for semantic analysis of movement behaviours.

Related work

There are a wide variety of techniques and algorithms that can be used for visual representation of STGs. Much research has been done on graph drawing in general, see Gibson et al. (2013) for a survey, and on visualisation of state transition graphs in particular (van Ham et al. (2001), van Ham et al. (2002), Pretorius and van Wijk (2006)). The major problem addressed by the researchers is reducing visual clutter and edge intersections, especially in representing very large and complex graphs. We leave the problems of graph visualisation beyond the scope of our work, since our goal is to discuss the use of STGs in movement analysis on a meta-level, irrespective of specific graph layout and drawing methods.

Examples of deriving STGs from movement data exist in the literature. Blaas et al. (2009) use STGs to represent movement behaviours of animals. For this purpose, domain experts examine time series of movement characteristics, divide these time series into internally homogeneous segments, and interpret the segments in terms of the activities performed by the animals. These activities, signified by labels, are taken as the states for creating an STG. The sequences in which the activities occur are represented by transitions. In this way, an individual movement behaviour is represented by an STG.

von Landesberger et al. (2016) use graphs to represent collective movements in geographic space. The states of an STG correspond to areas in space, and the transitions represent collective movements between these areas. The weight of a transition is proportional to the number of people that moved from the origin area to the destination area. Such graphs are built for different time intervals, i.e., in our terms, collective behaviours in these time intervals are represented by a chronological sequence of STGs. Such a sequence can also be treated as a dynamic graph, i.e., a graph where the nodes and/or edges appear, disappear, or change their weights (or other attributes) over time. The methods for visualisation and exploration of dynamic graphs have been comprehensively surveyed recently (Beck et al. (2014), Archambault et al. (2014), Hadlak et al. (2015)). von Landesberger et al. (2016) group similar graphs corresponding to different time steps into clusters and then study the distribution of the clusters over time and the differences between the clusters.

Andrienko et al. (2013d) and Andrienko et al. (2015) extract repeatedly visited places from trajectories of humans and assign labels to them according to inferred semantics of the places, e.g., 'home', 'work', 'shopping', etc. These semantic labels are taken as the possible states of the individuals. The state sequences are summarised into representations called "semantic space maps", which are in essence state transition graphs. The STGs are used for representing behaviours of singular or multiple individuals.

The use of STGs for representing and analysing multiple behaviours is not a general practice. A more common approach is to deal with behaviours as chronological sequences of states or events and visually represent these sequences in timeline displays (Tufte (1983), Aigner et al. (2011)). States, events, or activities are represented by lines or bars positioned in the display according to their occurrence times, the lengths being proportional to the durations (e.g., Plaisant et al. (1996)). Behaviours consisting of multiple states are often shown as horizontal or vertical segmented bars where coloured segments represent different states (Vrotsou et al. (2009), Vrotsou et al. (2010)). Behaviours can also be represented, according to the cyclic time model, as segmented rings (Zhao et al. (2008)).

When the number of possible states (activities, event types, etc.) is large, they are grouped into a smaller number of higher level categories (Zhao et al. (2008), Monroe et al. (2013)). To visualise a large number of behaviours, data aggregation is employed. Thus, ringmaps proposed by Zhao et al. (2008) summarise behaviours of multiple people. In LifeFlow (Wongsuphasawat et al. (2011)) and its successor EventFlow (Monroe et al. (2013)), multiple sequences are aggregated hierarchically. EventFlow also provides a number of other tools for display simplification based on filtering and data transformations.

State transition graphs provide a much higher degree of simplification than timeline-based representations. However, STGs are not a suitable instrument for studying the temporal distribution of states and transitions and for considering sub-sequences consisting of more than two states. Blaas et al. (2009) propose an approach in which sub-sequences consisting of three or more states are represented by smooth curves linking the corresponding graph nodes. However, this approach has limitations regarding the lengths of the sub-sequences that can be represented in this way and the number of distinct sub-sequences that can be shown simultaneously. Hence, STGs are not a valid alternative but rather a complement to timeline-based representations.

The STG representation has its specific niche, namely, tasks focusing on frequencies or durations of states and transitions, when behaviours can be treated as Markov chains (Norris (1998)). Aggregation of multiple behaviours and, when necessary, reducing the number of considered distinct states by filtering and/or combining them into meta-categories are used with STG analogously to the other representations (Zhao et al. (2008), Wongsuphasawat et al. (2011), Monroe et al. (2013)).

Another possible representation for (multiple) behaviours is interactive Sankey diagrams (Riehmann et al. (2005), von Landesberger et al. (2012)), which can be seen as a hybrid between the timeline-based and graph-based representations. A display shows frequencies of state occurrences at different selected time steps and frequencies of step-by-step transitions between the states. This can be considered as a graph where for each possible state there are as many nodes as the number of selected time steps. This multiplication of nodes allows, in principle, preservation of state sequences and temporality; however, information about the sequences can be accessed only by interacting with display elements, where each interaction provides only partial information.

As an alternative to Sankey diagrams, where intersections of graph edges can be a problem, Zhao et al. (2015) propose a zigzag arrangement of multiple matrices and, additionally, techniques for visual comparison between two state sequences. Gleicher et al. (2011) classify the techniques supporting visual comparison into three general categories: juxtaposition, superposition, and explicit encoding. Comparisons by juxtaposition (side-by-side comparison) and explicit encoding (computing and visualising differences) are applicable to both node-link and matrix-based representations of STGs whereas superposition (overlay) is limited to matrix-based displays. Zhao et al. (2015) use explicit encoding and superposition with matrix-based representations while von Landesberger et al. (2016) use explicit encoding and juxtaposition with node-link representations. In both examples, differences are encoded by shades of two distinct colour hues.

As this overview shows, there is a large repertoire of visualisation and interaction techniques supporting data analysis with the use of state transition graphs. It is not our intention to propose yet another technique. Our goal is to describe in a general way how a representation of movement data in the form of STG can be utilised in movement analysis. We are not going to recommend which particular visualization and interaction techniques need to be chosen since a variety of choices are possible. Particular techniques have been used only to make illustrations and should thus be treated as mere illustrations of the generic argument we present in this paper.

Transforming data to state transition graphs

Building state transition graphs from state sequences

A *state transition graph* is a tuple $< V, E, A^V, A^E >$, where V and E are, respectively, graph vertices (nodes) and edges, and A^V, A^E are, respectively, attributes of the vertices and edges. The vertices correspond to possible states, the edges to transitions between the states, and the attributes include the absolute and/or relative frequencies of the states and transitions and may also include their absolute and/or relative aggregate durations and other attributes. An STG can be derived from one or more sequences of states, where the term "states" refers to any entities, events, activities, qualitative attribute values, etc. Two cases are possible: simple state sequences $(s_1, s_2, ..., s_k)$ or time-referenced state sequences $((s_1, t_1), (s_2, t_2), ..., (s_k, t_k))$, where $s_i \in SS$, $SS = \{S_1, S_2, ..., S_m\}$ being a finite state of possible states called *state space*, and t_j being time intervals, such that $t_j < t_{j+1}$ for each $j \in [1, k)$. In a case of multiple state sequences, their lengths may differ, while the elements s_i must belong to the same state space SS.

Given a state space $SS = \{S_1, S_2, ..., S_m\}$ and a set of state sequences SQ (which may, in particular, consist of one sequence), a graph vertex V_i is created for each $S_i \in SS$. The *absolute state frequency* for V_i is the number of occurrences of the state S_i in all member sequences of SQ. The *relative state frequency* is the ratio or percentage of the absolute frequency to the sum of the absolute frequencies of all states of SS, which equals the sum of the lengths of all sequences of SQ. When the sequences are time referenced, the *absolute total state duration* for a vertex V_i is the sum of the lengths of all time intervals associated with the occurrences of the state S_i. The *relative total state duration* is the ratio or percentage of the absolute total duration to the sum of the absolute total durations of all states.

A directed graph edge E_{ij} is created for each pair of states (S_i, S_j), $S_i \in SS, S_j \in SS$, including also the pairs (S_i, S_i). Pairs (S_i, S_j) and (S_j, S_i), where i≠j, are treated as different. The *absolute transition frequency* for E_{ij} is the number of times when S_i is immediately followed by S_j in a member sequence of SQ. The *relative transition frequency* is the ratio or percentage of the absolute transition frequency of this edge to the sum of the absolute transition frequencies of all edges. When the sequences are time referenced, for each pair of consecutive occurrences $(s_n, t_n), (s_{n+1}, t_{n+1})$ in a sequence, the duration of the transition from s_n to s_{n+1} is computed as the time difference between the beginning of t_{n+1} and the end of t_n. The *absolute total transition duration* for edge E_{ij} is the sum of the durations of all transitions from S_i to S_j that occurred in the member sequences of SQ. The *relative total transition duration* is the ratio or percentage of the absolute total duration to the sum of the absolute total transition durations of all edges. For simplification, the vertices and edges having zero frequencies may be removed from an STG. For further simplification of the graph topology, the analyst may decide to ignore also edges with very low frequencies (below a chosen threshold).

As is clear from the description, the transformation to an STG is equally applicable to a single sequence and to an arbitrarily large set of sequences. Furthermore, from a set of STGs with a common state space (e.g., a set of graphs representing individual behaviours), a summary STG can be built. The vertices and edges of the summary graph are the same as in the primary graphs. The absolute frequencies and durations for the vertices and edges are obtained by summing the respective values over the primary graphs. The relative frequencies and durations are computed from the absolute ones in the same way as for the primary graphs.

Summary STGs can be built for any subsets of a set of STG, in particular, for subsets obtained through querying, partitioning, or direct selection.

The STG representation is obviously scalable to both the number and the lengths of the state sequences but not scalable to the size of the state space. When the state space is very large, the size can be reduced by grouping similar or semantically related states into categories and using these categories instead of the primary states (Zhao et al. (2008), Monroe et al. (2013)).

Let SQ be a set of time-referenced state sequences and $T = [t_0, t_{last}]$ be the time span of the data, where t_0 and t_{last} are, respectively, the earliest and latest time moments recorded in the data. It is possible to build an STG not from the entire sequences but from their parts fitting within a chosen time interval $[t_1, t_2]$, $t_1 \geq t_0$, $t_2 \leq t_{last}$. This transformation can be done in two ways. First, it may be done on the fly for time intervals interactively chosen by the analyst. Second, the analyst may partition T into several intervals, and an STG is built for each interval, resulting in a time-referenced set of STGs, or, in other words, a dynamic STG.

So far, we have described how STGs are built from arbitrary state sequences, such as health records (Wongsuphasawat et al. (2011), Monroe et al. (2013)), activity diaries (Vrotsou et al. (2009)), or web site navigation logs (Zhao et al. (2015)). In the next subsection, we describe derivation of STGs from movement data specified as trajectories of moving objects.

Transforming movement data: general approach

A trajectory is a sequence of time-referenced position records $((l_1, t_1), (l_2, t_2), ..., (l_k, t_k))$, where $l_1, l_2, ..., l_k$ are spatial locations. This data structure is analogous to time-referenced state sequences discussed before. However, the number of distinct locations occurring in movement data may be very large. Moreover, locations are most often specified by coordinates in a continuous (e.g., geographic) space where the number of distinct locations is infinite. Hence, trajectories cannot be directly transformed to state transition graphs by simply treating the spatial locations as states. An intermediate transformation is necessary: to define a suitable state space consisting of a moderate finite number of distinct states and to convert trajectories into time-referenced sequences of states taken from this state space.

There are two possible approaches to transforming trajectories into state sequences: *place-oriented* and *property-oriented*. In the place-oriented approach, possible states are defined based on places (i.e., parts of space) visited by moving objects. The places, in turn, may be defined either by partitioning the space into regions or by selecting particular parts of space, called places of interest (POI). The first variant, which may be called *space partition-based*, creates a full coverage of the space, i.e., any location belongs to some place. In the second variant (*POI-based*), some locations may remain out of any place. The places specified in either way are given labels that can be treated as possible states. Depending on the analysis goals, the places may be given unique labels or more general labels specifying place categories according to their properties, ways of use, or other semantics. For example, Andrienko et al. (2015) describe extraction of repeatedly visited individual and public places from human mobility data followed by inferring the likely meanings of these places or kinds of activities performed there. As a result, the places receive semantic labels 'home', 'work', 'shopping', 'eating', etc., which may be taken as possible states of the moving individuals.

After defining and labelling the places, the spatial locations occurring in the trajectories are replaced by the labels of the places containing these locations, which transforms the trajectories into time-referenced state sequences. In case of POI-based definition of states, the trajectory points not belonging to any POI are dropped as not relevant to the analysis.

In the property-oriented approach, possible states are defined based on the properties of the movement, such as speed, acceleration, and/or movement direction. Examples of possible labels are 'stop', 'slow movement', 'fast movement', 'turn', etc. Labels may also indicate transportation modes: car, bicycle, train, bus, etc. Such labels can be obtained using existing methods for identifying transportation modes from raw trajectory data, e.g., Xu et al. (2010).

Transforming movement data: examples

We shall consider four example datasets: (1) 59,439 daily trajectories of 17,241 distinct cars that moved in Milan over one week (Andrienko et al. (2013a)); (2) trajectories of 35 cars used by employees of a company, a synthetic data set provided for the VAST Challenge 2014 (VC2014), Mini Challenge 2; (3) activities of 11,374 visitors of an amusement park during three days, a synthetic data set provided for the VAST Challenge 2015 (VC2015); (4) trajectories of 20 field players from one half of a football (soccer) game, tracked with a resolution of 25 positions per second, about 75,000 positions per player.

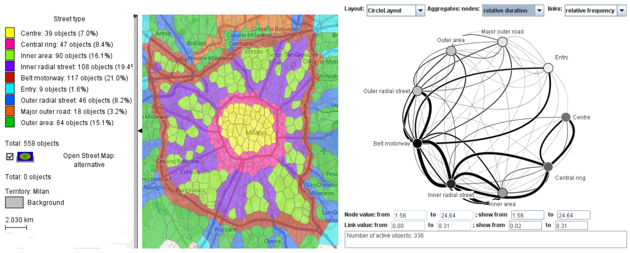

Figure 1. Left: Milan territory divided into areas, which are labelled according to the street types. Right: A summary STG for collective movements of the cars during 336 time intervals of the length of 30 minutes.

The purpose for transforming the Milan car trajectories to STGs is to explore the variety of car trips in terms of the categories of the streets and areas of the city traversed by the cars in abstraction from the specific locations and paths in the geographic space. We divide the territory of Milan into compartments and give semantic labels to these compartments according to the types of the streets they contain or the character of the areas (Fig.1, left). For 558 compartments, we have introduced 9 semantic categories, which make our state space. We substitute the positions in the trajectories by the labels of their containing compartments. We transform the resulting sequences to STGs in two ways: (1) an STG is built for each sequence, resulting in 59,439

STGs representing daily movements of individual cars; (2) the time span of the data (one week) is divided into 336 30-minutes intervals, and one STG is built for each interval, summarising the movements of all cars during that interval. The graph on the right of Fig.1 is the summary STG of the 336 time-based STGs. The shading of the graph nodes represents the relative durations of the states.

The first way of transformation is suitable for the exploration of the variety of individual car trips. Thus, the analyst can investigate the similarities and differences in the behaviors of the cars entering the city from the northwest and from the northeast (Fig. 2, top and middle). The second way of transformation allows the exploration of the collective movements of all cars during different time intervals. For example, the analyst may compare the collective movements in the morning and in the evening (Fig. 2, bottom).

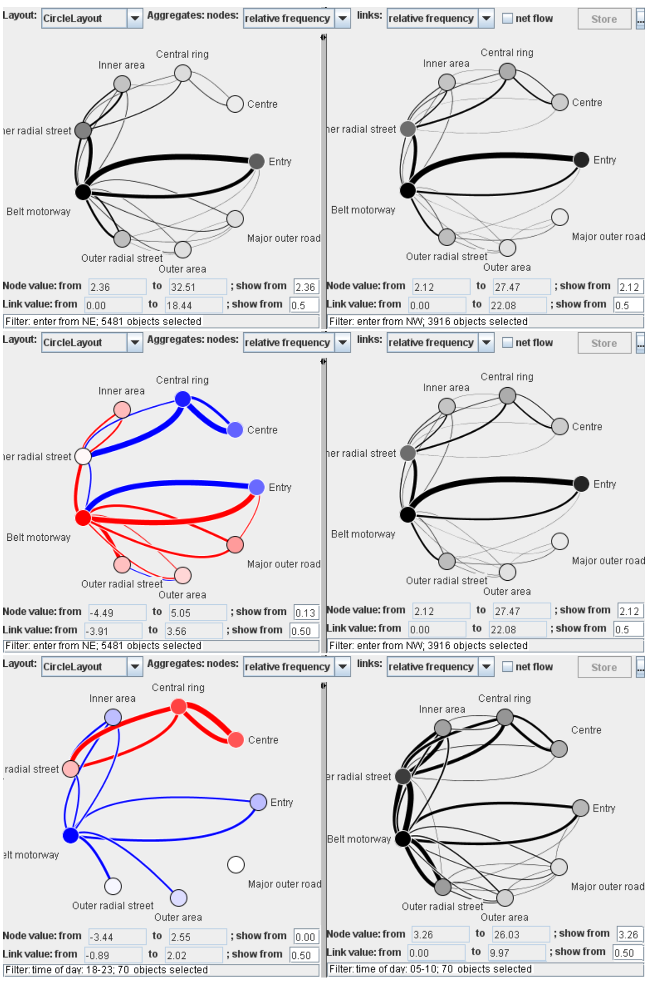

Figure 2. Top and middle: The behaviours of the cars entering the Milan territory from the northeast (left) are compared with those of the cars entering from the northwest (right). On the top, the respective summary STGs are exhibited for comparison by juxtaposition. In the middle, the differences are explicitly encoded. The red and blue colours show positive and negative differences, respectively. Bottom: The collective car movements in the evening (left) are compared with the collective movements in the morning (right) using explicit encoding of differences.

The VAST Challenge 2014 dataset contains trajectories of 35 cars and 5 trucks. The time span of the data is 14 days. Our illustrations are based on the car trajectories only. According to the scenario, the cars are used by employees of a company GasTech. In Fig. 3, top, there is a map of the fictitious city Abila where the GasTech employees live and work. The task of Mini Challenge 2 was to describe the usual daily behaviours of these people, which requires semantic interpretation and semantic analysis of the given trajectories. To accomplish the task, we first extracted repeatedly visited personal and public places from the trajectories and identified the likely meanings of these places based on the temporal patterns of the place visits (Andrienko et al. (2014), Andrienko et al. (2015)). The meanings were attached to the places as semantic labels. Then we divided the 35 complete car trajectories into daily trajectories, thus obtaining 471 daily trajectories. The latter were transformed into "semantic trajectories" (Andrienko et al. (2015)), that is, sequences of visits of the personal and public places represented by their semantic labels. The semantic trajectories were transformed into state transition graphs, where the states are the place meanings.

A summary STG of all daily sequences is shown in the lower part of Fig. 3. Apart from commonly understandable place meanings, the graph contains a node labelled "BFMO place". It stands for five specific places that were visited by

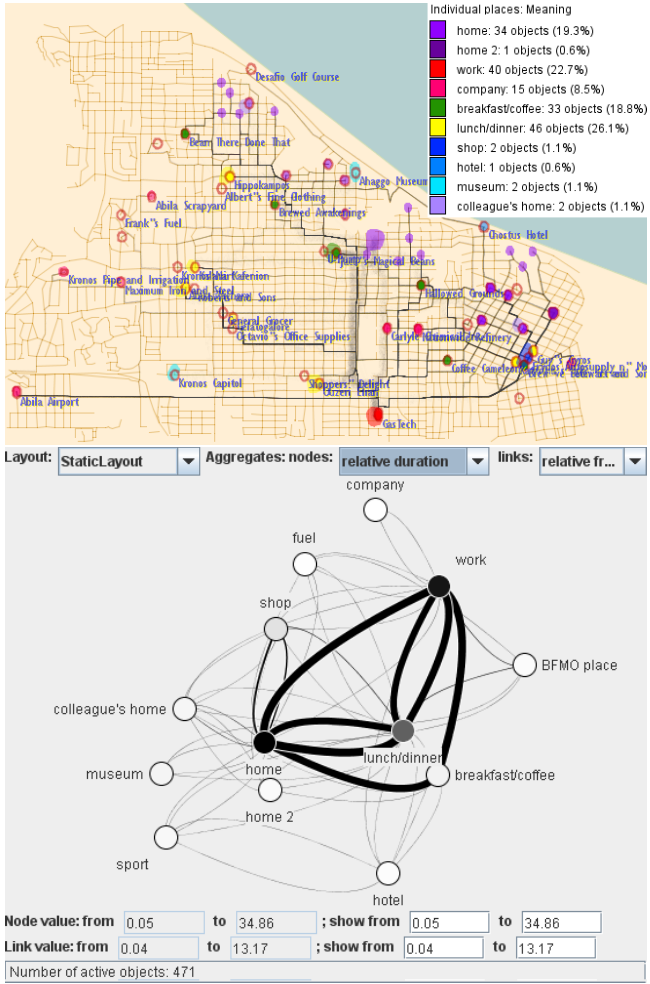

Figure 3. Top: A map of a fictitious city Abila (VAST Challenge 2014) with trajectories of employees of a company GasTech and personal and public places extracted from the trajectories. The personal places are coloured according to their meanings. Bottom: A summary STG represents the daily behaviours of the employees; the nodes correspond to the place meanings.

a particular group of four security employees, possibly, for secret meetings. “BFMO” is an abbreviation from the last names of these people.

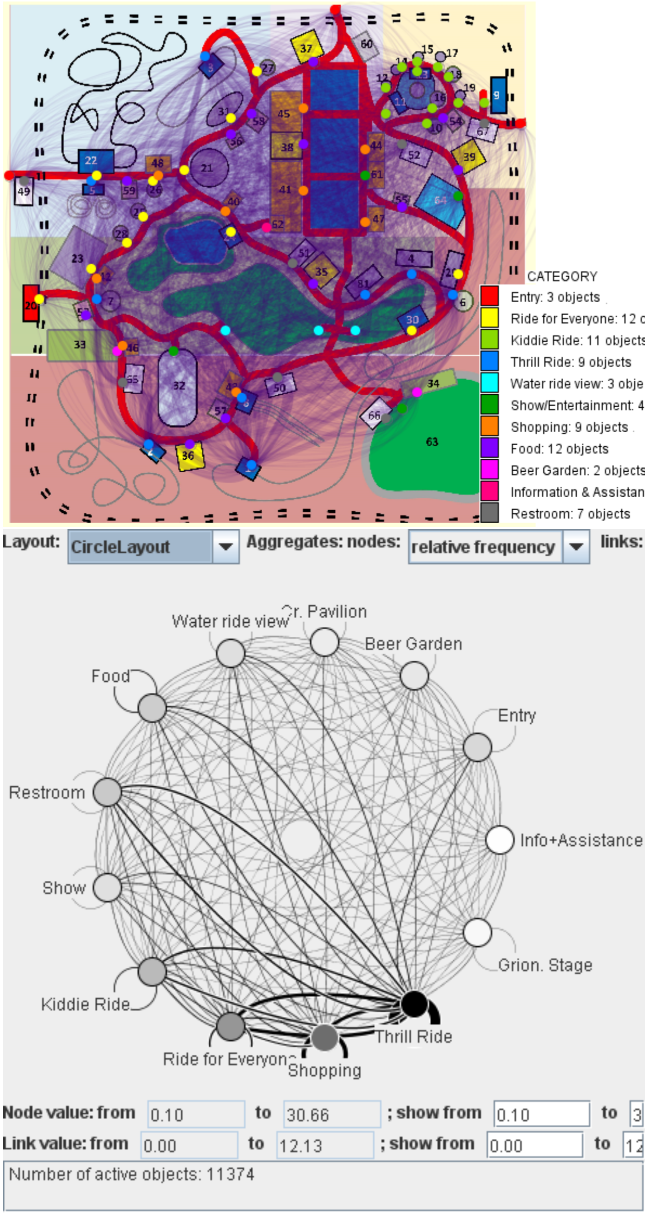

Figure 4. Top: A map of the amusement park (VAST Challenge 2015) with the attractions represented by dots coloured according to the semantic categories of the attractions. Semi-transparent curved lines in purple show aggregated movements of the park visitors between the attractions. Bottom: An STG represents aggregated movements of the park visitors between the semantic categories of attractions.

The VAST challenge 2015 dataset contains trajectories of amusement park visitors including time-referenced records of their checking in to various attractions. There are 73 distinct attractions grouped into 11 predefined semantic categories. The map in Fig. 4 (top) shows the park layout, the locations of the attractions, and the aggregated movements of the 11,374 park visitors between the attractions. In this example, our analysis goal is to detect unusual and, possibly, suspicious movement behaviours of park visitors. We transform the trajectories into state sequences using the POI-based approach, where the POI are the locations of the attractions. The state space consists of the 11 semantic categories of the attractions with an addition of the names of two specific attractions Creighton Pavilion and Grionosaurus Stage, which have a special role in the challenge. Fig. 4, bottom shows a summary STG of the individual behaviours of all park visitors. The shading of the graph nodes represents the relative frequencies of the states, and the widths of the edges are proportional to the relative frequencies of the transitions.

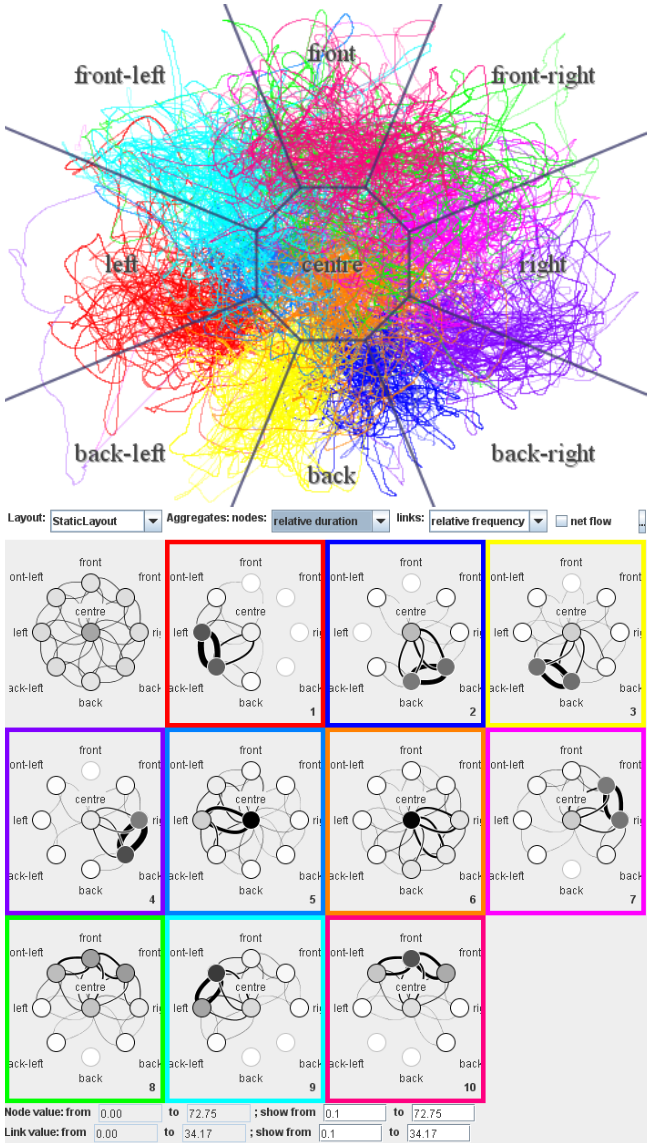

Figure 5. Top: Team space with transformed trajectories of field players (coloured curves) and division into 9 zones. Bottom: STGs of the movement behaviours of the field players of one team and the summary graph of the whole team.

In football, relative positions of field players in their teams have special meanings as they correspond to the roles and responsibilities of the players. Thus, the front positions belong to strikers, in the back are defenders, and players in the middle support the strikers or defenders and are responsible for passing the ball to the strikers during an attack and for pressing the opponents during the defense. In our example, we demonstrate how state transition graphs can be used for analysing the relative positions of players in a team irrespective of their specific absolute positions in the pitch. For this purpose, we first transform the players' trajectories to the "team space", similarly to Andrienko et al. (2013c). The team space is determined by the position of the team's centre and the direction of the team's attack. The team's centre is calculated in each time step as the mean position of all field players. The team's attack direction is the direction from the team's goal to the goal of the opponent team. The X-coordinate of a player in the team space reflects his relative position on the left or on the right of the team centre. The Y-coordinate reflects the player's relative position in the front or back part of the team with respect to the attack direction.

Having transformed the trajectories in this way, we divide the team space into 9 semantic zones, as shown in Fig.5, top, and replace the players' positions in the trajectories by the zone labels. The resulting state sequences are transformed to STGs. The lower part of Fig.5 shows the STGs of the field players of one team. The colours of the frames around the graphs are the same as those of the respective trajectory lines in Fig.5, top. The graph without a frame is the summary STG of the whole team. While the graph of the whole team is very symmetric, the graphs of the individual players reveal their specific relative positions and, hence, their roles in the team.

Advantages and limitations of state transition graphs

State transition graphs are particularly suitable for representing semantic abstractions of movement behaviours, as demonstrated by the examples in the previous section. A semantic abstraction enables consideration of the meaning or purposes of movement regardless of specific spatial locations where the movement takes place. Hence, the absence of specific spatial information is not a weakness but a strength of an STG representation. Obviously, a comprehensive analysis of movement requires that the abstract STG representation is used together with a representation containing specific spatial information, so that links between the semantic and spatial aspects can be established. Thus, to produce the illustrations in Fig. 2 (top and middle), we selected subsets of trajectories using their spatial representation.

Another possible representation for movement behaviour semantics is a state sequence. In essence, a state transition graph is an aggregation of one or more state sequences. The advantage of the STG representation over the representation by state sequences is its compactness, which is especially valuable in case of very long sequences. Besides, the STG represents the frequencies of the states and transitions and summary statistics of their durations (total, mean, median, etc.). This information is not available in the state sequence representation. One more advantage is scalability regarding the number of behaviours: an arbitrarily large number of behaviours can be summarized into a single STG. The state sequence representation does not allow such an easy aggregation of multiple behaviours. Displays of multiple state sequences may be quite complicated (Wongsuphasawat et al. (2011), Monroe et al. (2013)).

The STG representation becomes cumbersome when the number of distinct states is large, but this problem also exists for the representation in the form of state sequences. It is alleviated by grouping semantically related states into categories and replacing the elementary states by these categories (e.g., Monroe et al. (2013)).

A disadvantage of the STG representation over the state sequence representation is that state sequences consisting of three or more states are completely lost. When this information is important for analysis while the analyst wishes to utilize the advantages of STGs, it is necessary to combine the STG representation with another representation preserving the sequences. There are various possibilities for visual representation of state sequences. Blaas et al. (2009) include the sequence information in a graph display (node-link diagram) by drawing smooth curves connecting several graph nodes. This may increase display clutter and decrease its readability. An alternative is to represent sequence information in another view coordinated with the graph view. For example, state sequences can be visually represented as sequences of coloured bars in a timeline display (Plaisant et al. (1996)), as hierarchies (Wongsuphasawat et al. (2011)) and its successor EventFlow (Monroe et al. (2013)), or as sequences of edges in Sankey diagrams (Riehmann et al. (2005), von Landesberger et al. (2012)).

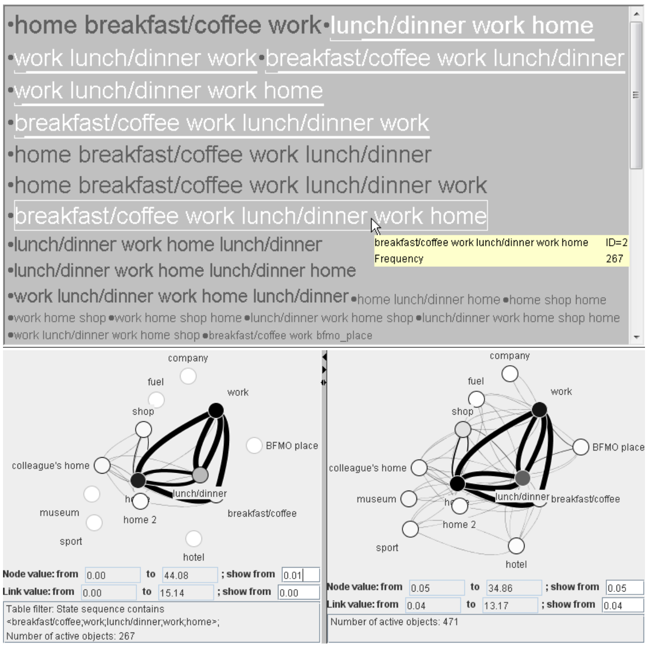

Figure 6. Top: A text cloud display represents state sequences of the length 3 and more. Bottom: The summary STG on the left represents the behaviours in which the selected sub-sequence of states occurs. The STG on the right summarizes all behaviours.

Yet a different possibility is demonstrated in Fig. 6. State sequences are represented by sequences of strings. Re-occurring sub-sequences of states are extracted using basic text analysis tools and represented visually in a text cloud display. The text-based representation can be used for interactive selection of behaviours including particular sub-sequences. The selected behaviours are shown in a graph view. The illustration in Fig. 6 has been produced based on the VAST Challenge data. The text cloud display shows re-occurring sequences of visits to places with different meanings. The mouse cursor points at the sequence "breakfast/coffee, work, lunch/dinner, work, home", which occurred 267 times in the daily behaviours of the GasTech employees. The graph view on the bottom left represents the summary STG of the 267 behaviours including this sub-sequence. It can be compared with the summary STG of all behaviours presented on the bottom right. Conveniently for analysis, the text cloud represents the relative frequencies of different sub-sequences.

Another disadvantage of state transition graphs may be the lack of temporal information. This is not always the case because movement data can be transformed into STGs representing collective behaviours by time steps. This was demonstrated in the previous section by example of the Milan data, which were transformed into 336 STGs each representing the collective movements during a particular time interval of 30 minutes length. This allowed us to select the collective behaviours according to the times when they occurred. In particular, we compared the behaviours that occurred in the morning to those having occurred in the evening (Fig. 2, bottom).

STGs encoding individual behaviours lack explicit time references. Accounting for the temporal aspect in analysis is possible by combining the STG representation with another representation preserving the original time references of the movement data. Temporal filtering can be applied to the time-aware representation. In response, the transformation to STGs is only applied to the data satisfying the filter. This possibility is demonstrated later by example of the football data: we compared the behaviours of the players before and after a goal.

A visual representation of an STG gives high prominence to frequently occurring states and transitions, which may be a disadvantage when the goal is to detect and investigate behaviour outliers. Another aspect of the problem is that a representation of a summary STG of multiple behaviours does not enable convenient access to individual behaviours. It is also not clear whether a state or transition has a high frequency because it occurs in many individual behaviours or because it occurs many times in a few individual behaviours. This is a drawback of the visual representation of multiple STGs in a summarized form, but this is not a limitation of the STG representation by itself. It is possible to display the information contained in the individual STGs in other ways, which may be more suitable for revealing outliers and for understanding how frequently the states and transitions occur in individual behaviours.

An example is demonstrated in Fig. 7, which has been produced using the data of the VAST Challenge 2015. In a bar chart display, the bars correspond to the transitions (not all transitions are simultaneously visible, but the display can be scrolled). In the upper image, the bar lengths are proportional to the counts of the individual behaviours where the transitions occur at least once. This shows which transitions occur in very few behaviours. These behaviours may be selected for close inspection by clicking on the respective bars. In the lower image, the bars are proportional to the maximal transition frequencies attained in individual behaviours. This reveals transitions that occur unusually frequently in some individual behaviours. These behaviours can be selected for detailed investigation. The current selection is marked in the bar display by dark gray shading.

Generally, the STG representation alone is typically not sufficient for analysis of movement data due to the limitations discussed above. STGs need to be combined with other representations of the same data preserving the spatial, temporal, and/or state sequence information. This requires proper coordination between analytical tools (including visual displays) working on the different representations.

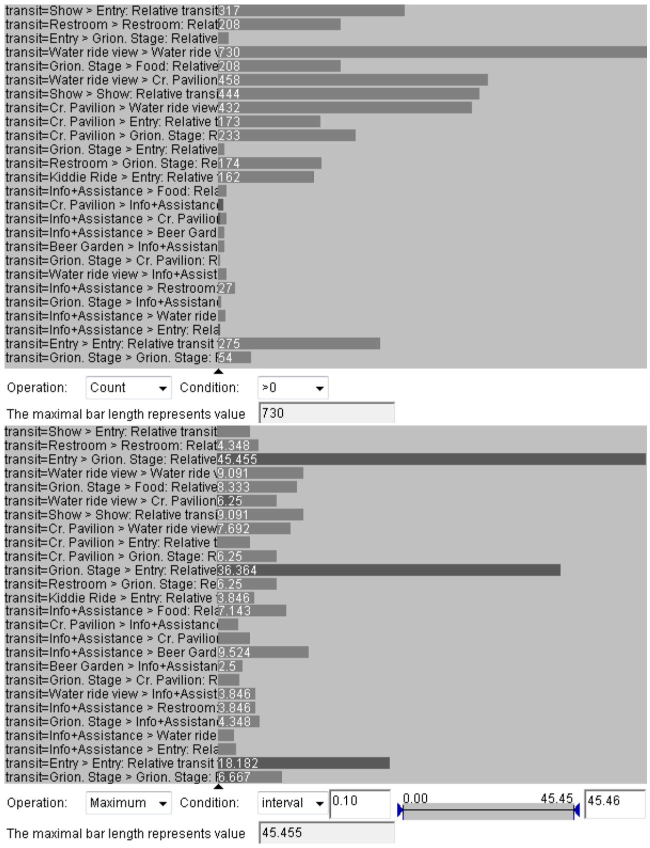

Figure 7. A bar chart display shows transition occurrences and frequencies in individual behaviours. Top: The bars represent the counts of the individual behaviours where the transitions occur at least once. Bottom: The bars represent the maximal frequencies attained by the transitions within individual behaviours.

Defining the STG tool

Our goal is to define a general approach to using state transition graphs for analysing movement semantics. We refer to general categories of tools and operations, which may be called “meta-tools” and “meta-operations”. Meta-tools may be defined by describing their properties and their behaviours in response to meta-operations. Meta-operations may be defined by describing their inputs and outputs and relationships between them.

Defining the properties of STG visualisation

An STG representation of movement data may comprise a large number of STGs, as can be seen in the earlier introduced VAST Challenge 2015 and Milan examples. It is unfeasible to visually represent all individual STGs. Consequently, the STG display must be able to represent STG data in a summarised form. As described earlier, multiple individual STGs can be aggregated in a summary STG with the same structure as the original STG. The STG tool must be able to create and visually represent summary

STGs for the whole set and various subsets of individual STGs. Thus, Figures 1-4 and 6 include such summary STGs of whole sets and subsets of behaviours. When the number of individual STGs under analysis is relatively small, analysts should also be able to see them separately, as in Fig. 5.

A key cognitive task in any analysis is comparison: noting similarities and differences is essential for generalization of observations, which contributes to building of a mental model of the studied phenomenon. Analysis with the use of STGs may require comparisons between individual STGs and/or subsets of STGs. Hence, the STG tool must support such comparisons using at least one of the approaches described by Gleicher et al. (2011): juxtaposition, in which two or more STGs are shown side by side, superposition, in which components of two STGs are drawn together (Zhao et al. (2015)), and explicit encoding, in which differences are represented by visual properties of the display elements standing for the nodes and edges of the graphs.

The way(s) to support comparisons need to be chosen together with the approach to graph visualisation. In principle, STGs can be visually represented using any currently existing or conceivable techniques for graph layout and graph drawing. Some of the possible methods for visual comparison support may have limited applicability regarding the graph visualisation methods. Thus, superposition is more suited to matrix-based representations of graphs than to node-link diagrams. Explicit encoding of differences by colours is not applicable to visualisations where colouring is used for other purposes. Juxtaposition is applicable to any representation, but it is the weakest support requiring analysts to repeatedly move their attention focus from one graph to another and to memorise the appearance of each graph under comparison.

In our illustrative implementation, graphs are represented as node-link diagrams, and comparisons are supported by juxtaposition and by explicit encoding using colours, as in Fig. 2. However, this is just one particular choice out of many possibilities. We refrain from discussing its strengths and weaknesses with regard to all others since our goal is to define the STG visualisation in a general way, i.e., as a meta-tool. Summarising this section, we state that the meta-tool for STG visualisation has the following properties:

- It visually represents state transition graphs using any graph layout and graph drawing techniques.
- It is able to aggregate multiple STGs into a summary STG and visually represent the summary STG.
- It also provides an opportunity to see selected individual STGs.
- It is able to show two or more STGs side by side, to support comparisons by juxtaposition.
- It includes at least one additional technique for supporting comparisons, i.e., superposition or direct encoding of differences.

Defining the behaviour of the STG tool

As discussed earlier, the STG representation may need to be used together with other representations of the same data. This means, in particular, that a visual display showing the data in the form of STG needs to be coordinated with other kinds of displays using suitable coordination mechanisms, such as consistent response to dynamic querying, brushing, grouping, sorting, and other interactive operations (Roberts (2007)).

Interactive operations are used not only for display coordination but also as means for data exploration and analysis. As we noted earlier, comparison is a fundamental operation that is necessary for deriving a general conception of a studied phenomenon. To perform a comparison in the course of data analysis, the analyst needs to define the portions of data to be compared. Regardless of possible implementation specifics, there are three basic approaches to choosing data portions for comparison: *querying*, *partitioning*, and *direct selection*.

Querying means that the analyst specifies some criteria differentiating data items of interest from other data and uses a tool that applies these criteria to the available data and extracts the requested data items. The extracted subset of data may be then compared with the remaining data, the whole dataset, or with results of other queries. Partitioning means dividing the whole dataset into two or more subsets, which are then compared. Direct selection means that the analyst chooses data items by interacting with their visual representations, without specifying any explicit selection criteria. According to these three approaches, the following common meta-operations (i.e., classes of interactive operations) can be defined:

- **Querying, or filtering**. Taking a set of data items as input, this meta-operation checks each data item against one or more query conditions and returns the subset of the data items satisfying the conditions.
- **Partitioning, or grouping**. This meta-operation divides a set of data items into two or more subsets, or item groups. The output is a specification of the group membership of each item, e.g., the label of the group it belongs to.
- **Direct selection, or marking**. This meta-operation assigns one of the two states, 'selected' ('marked') or 'not selected' ('not marked'), to each data item.

Please note that we disregard particular ways in which these operations may be accomplished. A variety of examples exist in the literature. A famous tool supporting querying based on attribute values is Dynamic Query (Shneiderman (1994)). Types of query tools that support exploration of movement data are described by Andrienko et al. (2013b). The meta-operation of direct selection is most commonly implemented as brushing (Becker and Cleveland (1987), Martin and Ward (1995)), in which the user selects and deselects data items by mouse clicking or dragging over their visual representations. This way of selection is also applicable to aggregated representations of multiple items, such as bars in histograms (Spence and Tweedie (1998)). Data items may also be marked in other ways. Thus, Bouvier and Oates (2008) propose a technique called "staining": display elements representing moving objects in an animated display become automatically marked when they get in touch with static or dynamic objects that have been previously chosen or created by the analyst. Partitioning of data items is often achieved through clustering (Aggarwal and Reddy (2014)), or partitions are defined based on values of categorical variables or by dividing ranges of numeric attributes into several intervals.

Irrespectively of how these meta-operations are implemented, let us define how an STG visualisation tool needs to react to them. We assume that an STG display originally presents a summary STG for all available data, the data being multiple individual STGs corresponding to different moving objects and/or different time intervals of movement. In parallel, there may be tools working on other representations of the same data, e.g., as trajectories, time series of movement-related attributes, or spatial events (Andrienko et al. (2013b)). Meta-operations may be applied to any representation of the data and performed by means of any tool. We assume that the results of the meta-operations are propagated to all other data representations, including the STG representation, and all other tools, in particular, to the STG display.

Behaviour in response to querying. The result of querying is a subset of the original set of STG. The STG display obviously needs to represent a summary STG of this subset, which requires the aggregation operation to be applied to this subset. It is not appropriate to just replace the previously shown STG by the summary STG corresponding to the result of the querying. Such a behaviour of the display does not allow the user to compare the result of querying with the previous state and to note changes. Therefore, the STG display should juxtapose the STG graph of the query result with the previously seen graph for enabling comparisons, for example, as shown in Fig. 2, top.

When a new meta-operation of querying is fulfilled, the result should be presented in a way consistent with presenting the result of the first operation. This means that the display needs to show the graph summarizing the last querying result and the summary graph of the previous querying result. For example, in our illustrative implementation, the graph that was previously shown in the left panel is moved to the right panel, and the new graph is put in the left panel. The graph that was earlier shown in the right panel is removed from the display, to avoid increasing the display size and complexity. However, it is appropriate to keep this graph in the internal storage and allow the user to select it at any time for comparison with other query results. This can be implemented in various ways. One possible strategy is to automatically store all graphs before they are removed from the view. The stored graphs need to be automatically labelled (annotated), e.g., by describing the query conditions applied to the data. Another possible strategy is to store a graph upon an explicit user request. In this case, the user may wish to supply his/her own label or annotation; still, the tool can help the user by proposing a default label/annotation. The former strategy may result in storing a large number of graphs that are not interesting to the user, and it may be difficult to the user to find interesting graphs among them. Hence, the latter strategy may be preferred, but it may be user- and task-specific.

Instead of storing summary graphs resulting from different queries, it is possible to store the query settings and reproduce the corresponding graphs on demand. This approach is more economical with regard to memory use but requires more time for graph re-generation. Which approach to choose when implementing an STG tool, depends on the expected number and sizes of graphs and available resources in terms of memory capacity and CPU speed.

The user should be able to control the way in which the STG tool reacts to a sequence of filtering operations. The tool behaviour we described so far is to show each time the result of the last operation and the result of the immediately preceding operation. Another possible behaviour is to keep a fixed reference view and juxtapose the result of the last operation with this view whereas the result of the previous operation is removed from the display. Depending on the analysis goals, the user may prefer this or that behaviour and should be able to choose between them. In both cases, either the graphs that are removed from the view are automatically stored, or the user has the possibility to store any graph for later review and comparison. In the mode of fixed reference view, the user should be able to choose any of the previously stored graphs to be used as a reference graph.

The rationale for showing two graphs in the STG display is to enable comparisons by juxtaposition. We have stated earlier that the STG tool should include at least one additional technique for supporting comparisons. Figure 2 includes examples of using direct encoding of differences. A reference graph for comparison can be chosen by direct manipulation, e.g., by clicking on a graph representation in the display. In response, the attribute values associated with the nodes and edges of the reference graph are subtracted from the attribute values of the corresponding nodes and edges of all other currently visible graphs. Positive and negative differences are encoded by shades of two distinct colour hues, e.g., red and blue. The use of an additional comparison-supporting technique is especially helpful when the differences are not obvious.

To summarise, an STG tool should behave in the following way in response to meta-operations of querying:

- Apply aggregation and build a summary STG for the data subset resulting from the querying operation.
- Juxtapose the summary STG of the last querying result with the STG that has been shown immediately before the querying or with a fixed reference STG selected by the user.
- Be able to store summary STGs representing results of different querying operations, either automatically or upon explicit user's request.
- Allow the user to choose any previously stored summary STG as a reference.
- Additionally support comparisons between STGs by superposition or direct encoding of differences.

Behaviour in response to partitioning. An appropriate way to present results of partitioning for viewing and analysis is to show summary STGs for the data partitions obtained. Each partition is a subset of individual STGs. To obtain the summary STGs, the STG tool must apply the aggregation operation to each subset. The summary STGs are juxtaposed within the display to enable comparisons, i.e., the technique of "small multiples" is applied (Tufte (1983)). This can be extended to a special case when each partition consists of only one individual STG, as in Fig. 5. In this case, no aggregation is done but the individual STGs are shown. In addition to the STGs of the partitions, the summary STG of the whole set of individual STGs may be included, allowing to compare the properties of each partition with the average properties over the whole dataset. As in the case of filtering, comparisons are supported by combining juxtaposition with superposition or direct encoding of differences.

Visual representation of data partitions often involves assignment of unique colours to the partitions, so that these colours are used for painting display elements representing partition members or the partitions as wholes. Thus, this is a frequently used technique in representing results of partition-based clustering. Consistent use of colours assigned to data partitions in multiple displays is one of coordination mechanisms allowing the user to link information pieces from the different display. Hence, when the STG display represents colour-encoded results of data partitioning, it must also show the colours of the partitions. A suitable approach is to enclose the graphs representing the partitions in coloured frames.

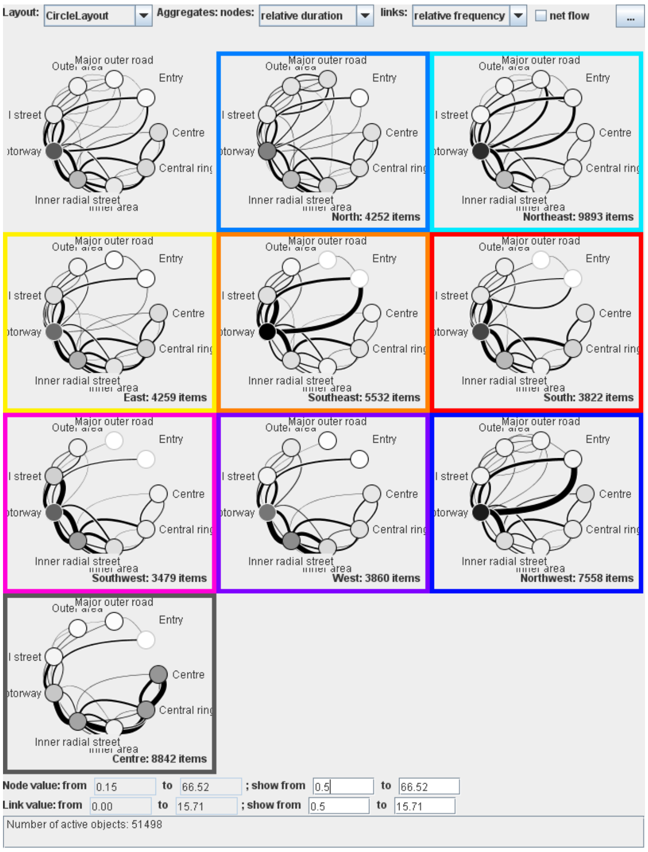

Figure 8. The Milan car trajectories have been partitioned into 9 classes according to the areas of their destination. The STG display represents the summary STGs of the classes of trajectories. The frames around the graphs are painted in the colours assigned to the classes.

As an example, the STG display in Fig. 8 reflects the results of partitioning of the set of Milan car trajectories into 9 classes depending on the geographic areas of their destination: centre, north, northeast, east, and so on. The partitioning has been performed by means of another tool operating on a geographic representation of the car trajectories. The result have been propagated to the STG display as a set of pairs (trajectory identifier, class label) together with a set of pairs (class label, colour). Since the individual STGs obtained by transforming the trajectories have the same identifiers as the trajectories they originate from, the STG tool was able to aggregate the STGs by the classes. The summary STGs of the classes are shown in a "small multiples" view. The display compartments containing these STGs have frames painted in the colours of the classes. The class labels and member counts are shown by texts in the lower right corners of the compartments. The graph without a frame in the top left corner of the display is the summary graph of the whole set of individual STGs. In this example, the shading of the nodes encodes the average relative durations of driving in the streets of different types, and the widths of the link lines encode the average relative frequencies of the transitions between the street types. Differences between the classes are easily seen.

In data analysis, it may be necessary to compare results of different data partitioning operations. In particular, when cluster analysis is performed, the analyst often does not know in advance what parameter settings to choose. The analyst may apply clustering with different settings and study differences between the corresponding results in order to understand the impact of the parameters and to eventually choose suitable settings. To support comparisons between results of different partition operations, the STG tool may divide the display into two parts. Each part shows the summary graphs of the data partitions resulting from one partitioning operation.

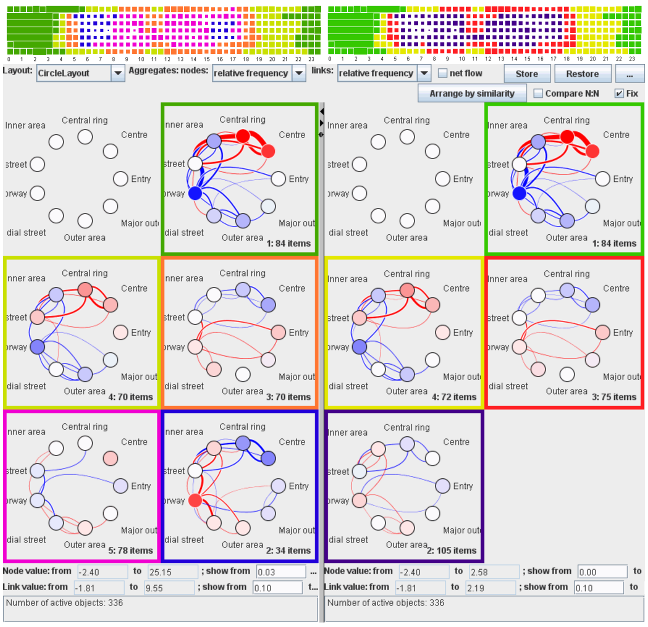

Figure 9. The STG display of the collective behaviours of the Milan cars represents results of two clustering operations that have produced 4 clusters (right side) and 5 clusters (left side). Clustering has been applied to traffic situations in 30-minute time steps and resulted in grouping the time steps into clusters by similarity of the traffic situations. The calendar views (top) show the distribution of the clusters over the week, and the STG view shows summaries of the collective behaviours corresponding to the clusters.

An example is presented in Fig. 9. Here we also use STGs derived from the Milan car trajectories, but in this case we use the STGs representing the collective behaviours of the cars by 30-minutes time steps. Partitioning by means of clustering has been applied to traffic situations (Andrienko et al. (2012), Andrienko et al. (2013b)) in the geographic space. A traffic situation in a time interval is described in terms of the volumes and characteristics of the traffic flows between areas in space. In our example, the same 30-minutes time intervals have been used for the creation of the STGs and for the characterisation of the traffic situations.

Clustering by k-means with $k = 4$ (right) and $k = 5$ (left) has been applied to the traffic situations, which results in grouping the time intervals by similarity of the situations. The calendar views in the upper part of Fig. 9 show the distribution of the clusters over a week. The rows correspond to the days, from Sunday on the top to Saturday in the bottom, and the columns to the 30-minutes time intervals within the days. Square marks in the cells are painted in the unique colours assigned to the clusters, the square sizes being inversely proportional to the distances of the cluster members to their cluster centres (hence, larger squares correspond to more typical cluster members).

In the lower part of Fig. 9, the STGs representing the collective behaviours of the cars by the 30-minutes intervals have been aggregated by the time clusters into summary STGs. The right and left parts of the STG display show the clustering results for $k = 4$ and $k = 5$, respectively. The result of each clustering is shown by small multiples, analogously to Fig. 8. Since in this case the differences between the STGs are hard to perceive from juxtaposed representations, explicit encoding of differences is applied as additional support to comparisons. The STGs of the clusters are compared with the STG of the whole dataset. The representation of the node and link characteristics of the overall STG is muted, to allow more expressive representation of the differences in all other graphs. The user can interactively select any other STG as a reference for comparison and, when necessary, mute the representation of graphs with extreme values of the node- and/or link-related attributes, so that narrower value ranges of the attributes could be mapped onto the available value ranges of the visual variables 'colour darkness' and 'line width'.

When clustering with different parameter settings is applied to the same data, it is not guaranteed that ordering and labeling of the clusters are consistent between results of different runs of the clustering tool. To facilitate comparisons between results of two runs, the STG tool should be able to arrange the STGs of the clusters within the left and right panels of the display so that similar STGs appear in the same positions on the left and on the right. Arrangement by pairwise similarity of the component graphs in the two panels has been applied in Fig. 9. Next to the overall STG in each display half, the STG of cluster 1 is put since this cluster is exactly the same in the two clustering results. It is followed by clusters 4 and 3, which changed slightly when $k = 4$ was replaced by $k = 5$. The last in the right panel is the STG of cluster 2, which changed the most significantly. Its counterpart in the left panel is the STG of cluster 5, which is more similar to the STG of the earlier obtained cluster 2 than the STG of cluster 2 returned by the clustering with $k = 5$. The latter STG is put in the last position in the left panel since it is the most dissimilar to all graphs contained in the right panel.

To accomplish the arrangement by graph similarity, each graph in the left panel is compared to each graph in the right panel. The dissimilarity measure is the sum of squared differences between the weights of the corresponding nodes and edges. The weights are the values of the currently shown attributes, i.e., absolute or relative frequencies or durations, standardised to the scale [0..1]. To arrange the graphs, the closest pair of graphs is repeatedly chosen, and the graphs are put in the next free slots in the left and right panels, starting from the top left corner.

Any of the multiple graphs can be interactively selected by the user as a reference for comparison. As a result, the differences of all other graphs with regard to the reference graphs are explicitly encoded by colour hues, node shades, and line widths, as described earlier. This allows, in particular, exploring fine differences between counterpart graphs from different panels.

When multiple partitioning operations are applied one after another, the behaviour of the STG display must be consistent with the behaviour in response to multiple querying operations. That is, one of the display panels needs to show the result of the last partitioning, and the other display part shows the result of either the directly preceding partitioning or a fixed reference partitioning. Like with querying, collections of STGs representing results of different partitioning operations are either stored automatically or may be stored by explicit user's requests and later chosen for reviewing and comparison.

To summarise, an STG display has the following behaviour in response to meta-operations of partitioning:

- It applies aggregation and produces a summary STG for each partition.
- It shows the STGs corresponding to the different partitions in a "small multiples" view.
- When the partitions are distinguished by colours, these colours are shown in the display compartments corresponding to the partitions.
- Comparisons between summary STGs of different partitions are supported, additionally to the juxtaposition, by superposition or direct encoding of differences. The user may interactively select any STG as a reference for visual comparison.
- Results of two different partitioning operations may be juxtaposed in the STG display. The STGs within the display parts corresponding to these different operations may be arranged in the order of pairwise similarity.
- The behaviour regarding multiple partitioning operations is consistent with the behaviour regarding multiple querying operations.
- Collections of STGs representing results of different partitioning operations may be stored for later reviewing and comparing with other results. The storing strategy (automatic or by explicit request) is the same as in the case of querying.

Behaviour in response to direct selection. The meta-operation of direct selection produces results similar by their structure to results of partitioning. Indeed, data items are partitioned into two subsets, 'selected' and 'not selected'. Still, there are differences, which require the STG tool to behave differently in response to selection than in response to partitioning. First, selection by direct manipulation of display elements is a quick operation the results of which can be easily changed by the user. It would be wrong to represent such transient results in the same way as, for example, results of clustering, which typically takes more time. Second, direct selection can be applied to previously partitioned data. The partitioning remains the same when the selection changes. Hence, it would be wrong to replace a result of partitioning shown in an STG display by a result of direct selection and further react to selection changes while the view of the partitioning result is lost. It is more appropriate to present results of direct selections separately. They should be shown in a visually distinct way, to preclude user's confusion. To fulfill this requirement in our illustrative implementation, we chose to show transient results of direct selections in a separate window, which appears only when a selection is made and disappears when everything is deselected.

When the user selects a subset of data items, it is appropriate to enable the user to compare this subset with the subset of remaining items. To achieve this, the STG tool must react to selection by juxtaposing two summary STGs: the STG for the selected items and the STG for the remaining items. Comparisons need to be additionally supported in the same way as it is done for results of filtering and partitioning.

An example of STG behaviour in response to direct selection is demonstrated in Fig. 10. We use interactive histograms to explore the characteristics of the Milan car trajectories and relate them to the behaviours in terms of the street types. A histogram of the track lengths demonstrates a long-tail distribution, i.e., very long trajectories exist, but they are relatively few. We interactively select the "long tail" in the histogram display (Fig. 10, A). The STG tool creates a supplementary window with a summary STG of the selected trajectories and a summary STG of the remaining trajectories (Fig. 10, B). The former STG is enclosed in a black frame since black colour is used to mark selected items in the histogram and in other displays. The link widths in the STGs represent the transition frequencies and the darkness of the nodes represents the relative durations of the states. We see that the longest trajectories (probably, made by taxis or delivery vehicles) spend most of the time in the city centre, and the most frequent transitions are between the centre and the central ring. Still, the motorway and inner radial streets are also quite frequently used. These vehicles mostly do not go to the outer sides of the belt motorway. In Fig. 10, C, we have cancelled the selection of the long trajectories and instead selected the leftmost bar of the histogram corresponding to the lengths below 5 km. Our expectation has been that short trips mainly connect places inside the city. However, the summary STG of the selected trips (Fig. 10, D) shows us that most of the time of the shortest trips was spent on the belt motorway. These trajectories did not go to the centre, and the most frequent transitions are from outer radial streets to the belt motorway and from the belt motorway to the inner radial streets. We also note asymmetry between the inward and outward flows.

In this example, selections were made by interacting with another display (a histogram) while the STG display only showed the results. Can the user interactively select subsets of data within the STG display? The answer is: not directly,

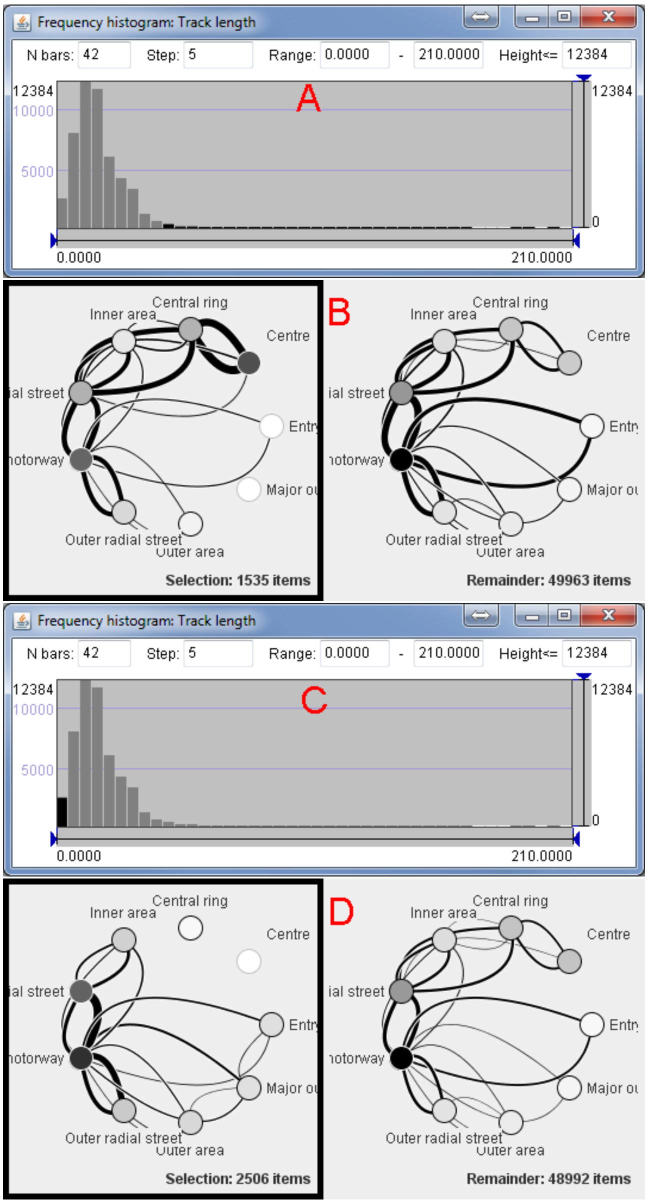

Figure 10. A frequency histogram display is used for direct selection of Milan car trajectories based on their length. The upper STG view (B) has appeared in response to selecting the trajectories with the length 45 km or more (A). The lower STG view (D) has appeared in response to selecting the leftmost bar of the histogram (C) where the track length is below 5 km.

for the following reason. In many display types, display elements represent either individual data items (e.g., dots in a scatterplot, polygonal lines in a parallel coordinates plot, trajectory lines on a map, etc.) or subsets of data items (e.g., bars in a histogram or aggregated trajectories on a map). Selection of such display elements can be naturally translated to selection of the corresponding data items. An STG display is different from these display types. The data items represented in an STG display are STGs; the display shows a summary STG of these STGs. The display elements represent not individual data items or subsets but the set of states and the set of transitions, which are common for all or many data items. Selection of a display element representing a state or a transition would translate to selection of all data items in which this state or transition appears. In most cases, these would be all or almost all data items, which is not useful.

A possible solution is to perform selections in two steps. In the first step, the user selects a state or a transition. In response, the STG tool displays a histogram of attribute values associated with the selected state or transition. For the histogram, the tool takes the values of the currently shown attribute, i.e., the absolute or relative frequency or duration. In the second step, the user interacts with the histogram to select data items according to the attribute values, similarly to what we did in Fig. 10 with an external histogram display. The histogram view once created by the STG tool needs to stay on the screen to enable changes of the selection and as a reminder to the user what data items are currently selected. The user may explicitly close the histogram view when it is not needed any more.

Let us summarise the definition of the behaviour of the STG tool in response to direct selection:

- The STG tool applies aggregation to the subsets of selected and remaining data items and presents the summary STGs of these subsets together to enable comparison by juxtaposition.
- Results of selection operations are shown separately and distinctly from results of querying and partitioning.
- Additional comparison-supporting techniques (superposition or direct encoding of differences) are provided as for querying and partitioning.

Examples of the use of state transition graphs in movement analysis

The capability of an STG visualisation tool to react to analytical meta-operations allows integration of this tool into a visual analytics system for movement analysis. According to Andrienko et al. (2013b), transformations from one data type to another may be useful and often even necessary in analysis of movement data. The book describes three major representations of movement data, trajectories, spatial events, and spatial time series, along with the possible transformations between these representations. By combining analytical tools working on these different representations, movement data can be analysed more comprehensively. In this paper, we introduce additional representations of movement data: state sequences and state-transition graphs. These representations support abstraction from the geographic space and allow semantics-oriented analysis and reasoning and finding common behavioural patterns in geographically and temporally unrelated movements. Owing to this high level of abstraction, the STG representation can be a good complement to the space- and time-based representations. For such complementary use, the STG tool must be responsive to analytical operations performed on other representations of the same data, as described in the previous section. Reciprocally, there must also be a possibility to perform analytical operations on the STG representation of movement data and propagate the results to other tools working on other representations.

The STG representation allows analytical meta-operations of querying, partitioning, and direct selection based on the attribute values associated with the graph nodes and edges. The attributes are the absolute and relative frequencies and durations of the states and transitions. The general way of using these attributes is the same as for any numeric attributes. In this section, we shall demonstrate examples of movement analysis where analytical operations are applied to the STG representations.

Discovering typical daily behaviours of GasTech employees

The analysis task in VAST Challenge 2014 is to reveal and describe the typical daily behaviours of the GasTech employees. Density-based clustering is a suitable tool for detecting what is typical (i.e., frequently occurring). Density-based clustering (Ester et al. (1996)) finds groups of similar objects and puts them in clusters. Objects that are not sufficiently similar to others are labelled as "noise". In our example, we apply a density-based clustering algorithm OPTICS proposed by Ankerst et al. (1999). The results of OPTICS depend on two parameters: the maximal distance (dissimilarity) threshold D and the minimal number of neighbours (i.e., similar objects) N. We apply OPTICS to the vectors of the relative transition frequencies of the individual STGs. The distance (dissimilarity) measure is the average difference between the transition frequencies in two STGs.

The right panel of the STG view in Fig. 11 displays the results of the OPTICS clustering with $D = 1.2$ and $N = 3$. OPTICS detected 10 clusters of similar STGs and labelled 68 STGs as "noise". The clusters have been assigned distinguishing colours; dark grey has been assigned to the "noise". The colours are used to indicate the clusters in the STG display: the summary STGs of each cluster is enclosed in a frame of the respective colour. The cluster labels and sizes are shown in the bottom right corners within the frames. The sizes of most of the clusters range from 4 to 52 items (i.e., individual STGs included in the clusters). The corresponding behaviours are quite clearly perceived from the summary STGs of the clusters.

However, there is one very large cluster, namely, cluster 1 (indicated by red colour) with 278 members. Its summary STG is shown in the last position in the right panel. Judging from the multitude of thin edges (i.e., infrequent transitions), the cluster has a high internal variance. We find it appropriate to refine cluster 1 by decreasing the distance threshold. This can be done interactively using the earlier obtained OPTICS output, without full re-clustering. So, we lower the distance threshold D to 1.1, which divides cluster 1 into four new clusters and adds 7 STGs to the "noise" category, as they are not sufficiently similar to others regarding the new distance threshold. Unfortunately, the cluster labels are not preserved after the refinement operation. The descendants of cluster 1 receive labels from 1 to 4, and the remaining clusters are labelled starting from 5.

To facilitate comparisons between the previous set of clusters and the new one, we apply the arrangement by similarity between the STGs in the two panels. As a result, the clusters that did not change are put at the beginning of the arrangement. We see that the clusters from 5 to 13 in the new cluster set correspond to the clusters from 2 to 10 in the previous cluster set. The summary STGs of the "noise" are

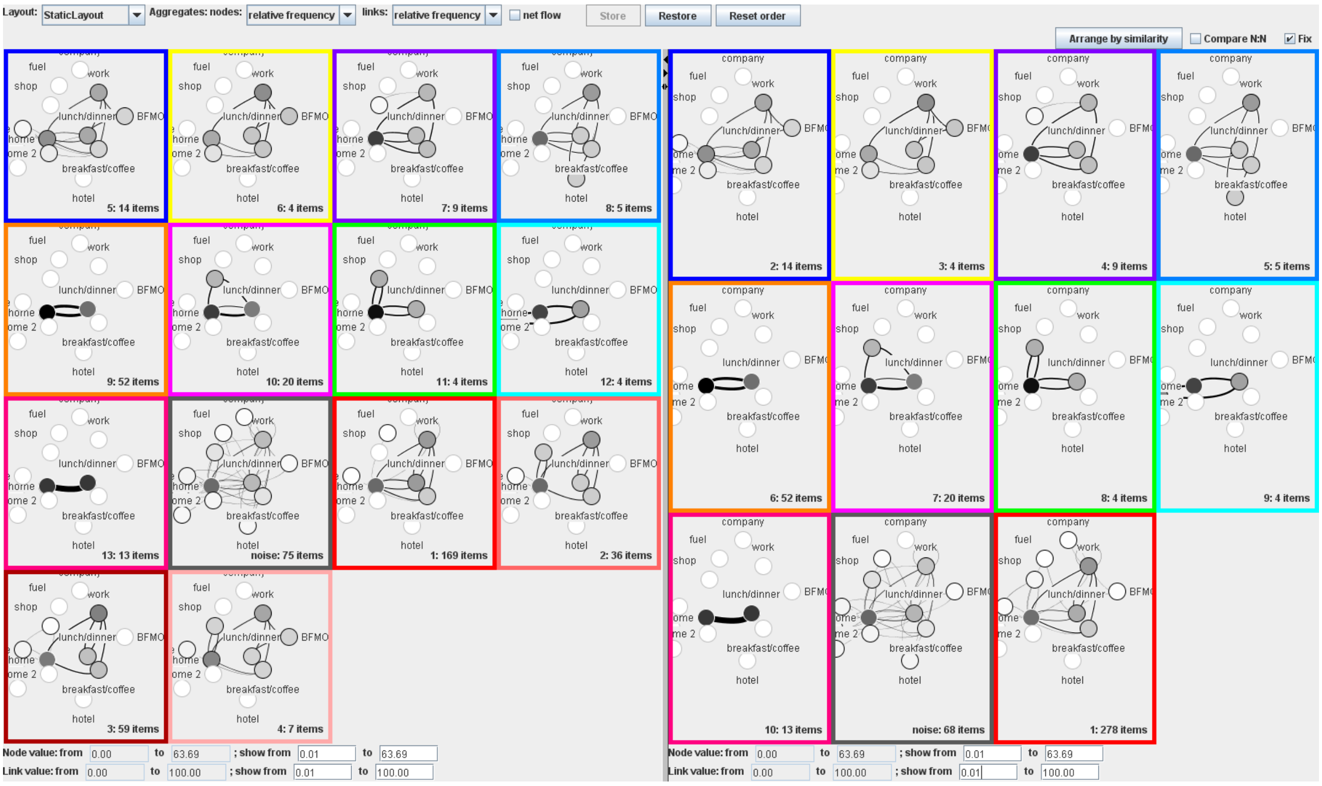

Figure 11. Analysis of daily behaviours of GasTech employees by iterative density-based clustering of state transition graphs.

similar in the two the cluster sets. Former cluster 1 is shown in the last position in the right panel because it has greatly changed. In the left panel, new cluster 1 is shown in the same position as the old cluster 1. Among the descendants of the old cluster 1, its STG is the most similar to the STG of the ancestor. The new clusters 2 to 4 are put at the end of the arrangement.

After the refinement, the STGs of all clusters (disregarding the "noise") are easily interpretable. The most typical daily behaviour is represented by cluster 1 consisting of 169 STGs. It includes movements from home to a place of breakfast or coffee, from the breakfast/coffee place to work, and from work to home. It also includes two-way movements between the work place and a place of lunch/dinner and between home and lunch/dinner. Although the transition sequence is not represented in the graphs, we can plausibly guess that people moved in the morning from home to work with an intermediate stop for having breakfast or coffee. At the lunch time, they went from work to a lunch/dinner place and then returned back to work. After coming home from work, they went for dinner to a lunch/dinner place and returned home after that. A coordinated text cloud display of state sequences, as in Fig. 6, confirms these guesses.

Clusters 2 to 4 exhibit variations of the archetypical behaviour represented by cluster 1. In cluster 2, dining is substituted for shopping. In cluster 3, there are no transitions from home to shopping or dining places. 11 behaviours include visits to a colleague's home; in the remaining behaviours, the people simply stayed at home after work. Cluster 4 consists of 7 behaviours that include visits to the BFMO places (i.e., places of meetings of security employees).

Similarly to cluster 4, cluster 6 also contains behaviours with visits to the BFMO places, but, unlike in cluster 4, there are no visits to shopping places. Visits to the BFMO places also appear in cluster 5; the remaining states and transitions in this cluster are mostly the same as in the archetypical behaviour (cluster 1). The behaviours in clusters 7 and 8 do not include transitions from work to lunch/dinner and back. The specifics of cluster 8 is the presence of two-way transitions between work and hotel. Clusters from 9 to 13, evidently, represent various behaviours during weekends, when people did not go to work.

This example demonstrates the application of partitioning to the STG representation of movement behaviour semantics, which facilitated the discovery and interpretation of typical behaviour patterns and their variations.

Detecting unusual behaviours of amusement park visitors

This example is based on the VAST Challenge 2015. Unusual behaviours in terms of extremely high or low frequencies or durations of some states and/or transitions can be detected using suitable displays of the node or link attributes, such as the bar display in Fig. 7 representing transition frequencies. We create similar displays showing the relative frequencies and durations of the states (i.e., attraction categories) in the VAST Challenge example and perform direct selection operations by clicking on bars representing unusually high or low frequencies or durations. Figure 12 demonstrates the results of several direct selections that have revealed outliers among the behaviours of the park visitors.

The graph on the top left of Fig. 12 presents the behaviours with extremely high frequencies of the state 'Entry' (64.3%). These behaviours have also extremely high frequencies of

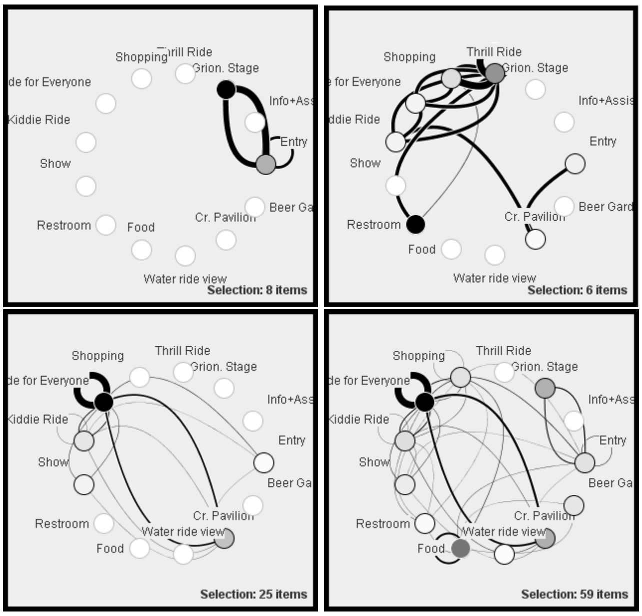

Figure 12. Summary STGs of interactively selected outlying behaviours of park visitors.

the state ’Grionosaurus Stage’ (35.7%). The summary STG tells us that our selection consists of behaviours of 8 people who were only at the entry and at Grionosaurus Stage and did not visit any other attractions. Obviously, these were people who, according to the VAST Challenge 2015 story, accompanied a special guest to dedicated shows at Grionosaurus Stage.

On the top right, we have selected the behaviours with unusually much relative time spent in restrooms (57.78%). The graph shows us that the only transition leading to the state ’Restroom’ originates from ’Thrill Ride’. We can guess that the people (6 persons) whose behaviours are selected might feel sick while attending Thrill Rides. We also observe that these people did not return to Thrill Rides after visiting restrooms. Moreover, there was only one transition from the state ’Restroom’, and it was a single person who moved to ’Shopping’. The remaining 5 persons did not go to any other attraction after visiting restrooms.

On the lower left of Fig. 12, there is a summary STG of the behaviours of 25 people who spent 50% or more of their time for attractions of the category ’Ride for Everyone’. The STG shows us that these people did not attend Thrill Rides, which was the most popular attraction category. On the lower right, we have selected all behaviours with no visits to Thrill Rides. There were only 59 people (out of 11,374) who did not go to Thrill Rides. This includes the earlier discussed group of 8 people who went from ’Entry’ directly to ’Grionosaurus Stage’ and then returned back. The remaining people were mostly interested in Rides for Everyone. There were also many visits to Creighton Pavilion, and much time was dedicated to the attraction category ’Food’.

This example demonstrates detection of behavioural outliers by applying direct selection to the STG representation of behaviours.

Visually comparing behaviours of football players

As we noted in introducing the football data example, the relative positions of players in their teams are indicative of their roles and responsibilities and, hence, semantically meaningful. In this section, we first investigate whether the two teams had similar formations and distributions of roles among the field players. We set the STG display so that each of the two panels shows the STGs of the players of one team (Fig.13). The left panel corresponds to the home team and the right panel to the away team.

We apply the arrangement of the graphs in the two panels by pairwise similarity and observe that there are both pairs of very similar and pairs of dissimilar players in terms of the relative positions and movements within their teams (Fig.13). More specifically, there were 6 pairs of players with very similar behaviours. The similarity-based arrangement has put the STGs of these players in the upper three rows of the left and right panels. We can clearly see the similarity of the behaviours of players 10 and 21, 1 and 12, 7 and 19, and so on. In particular, the strikers of the home team (10 and 8) behaved quite similarly to the strikers of the away team (21 and 20, respectively) . The pairs of STGs located closer to the bottom of the display are less similar. In the bottom row, we see that the home team had a central player who also frequently appeared on the left (number 5), whereas the away team had instead a central player with a tendency to the right (18). For the home team player who mostly played in the front-left position and less frequently in the middle-left (9), the counterpart in the away team (17) was more focused on the middle-left position and more frequently moved to the centre and back.

Next, we want to investigate whether any players changed their behaviours after the guest team had scored a goal on the 13th minute of the game. We apply the transformation to STGs separately to the parts of the player’s trajectories recorded before and after the goal and set the STG display so that the behaviours of the players of both teams after and before the goal are shown in the left and right panel, respectively. We arrange the multiple STG by similarity in order to reveal possible dramatic changes, such as some players having swapped their relative positions. We see that this did not happen: for each player, the graph position in the left and in the right panel is the same. However, the graphs of the players who changed their positions most significantly have moved to the bottoms of the left and right panels. Among these, there are three players of the home team (numbers 7, 8, and 10) and one player of the guest team (number 15) .

To clearly see the changes of the behaviours, we apply a special comparison mode N:N, in which the left panel portrays the differences of its multiple graphs from the corresponding graphs of the right panel, which keep their original appearance. A fragment of the display (the bottom part of it) is shown in Fig.14. The largest change happened to a striker of the home team (number 10; the STG is in the lower right corner), who shifted his position from the front and front-left to the front right. The player number 8 (the STG in a bright green frame), who initially tended to the centre, front, and front right, shifted more to the front

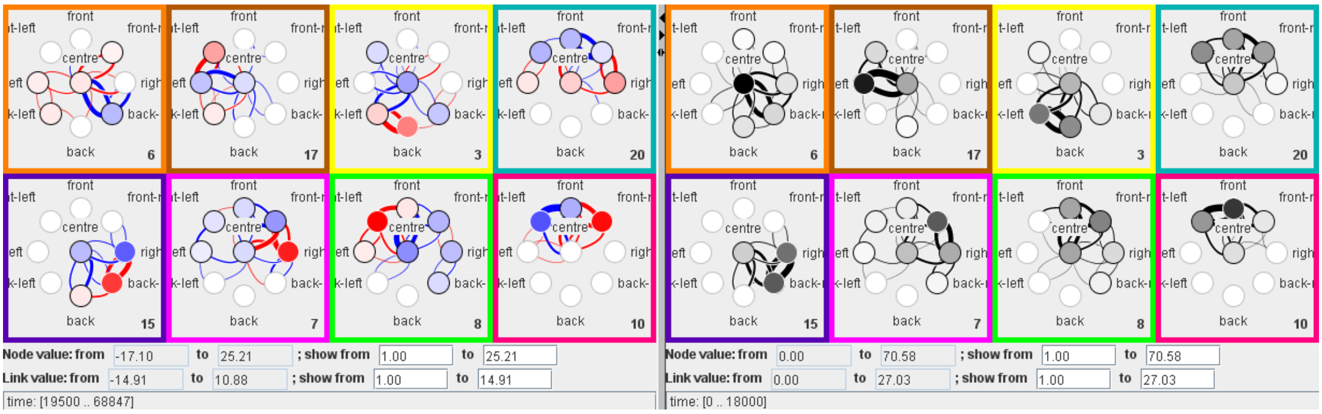

Figure 14. The left panel shows the changes of the behaviours of 8 players after a goal in comparison to their behaviours before the goal, which are shown in the right panel.

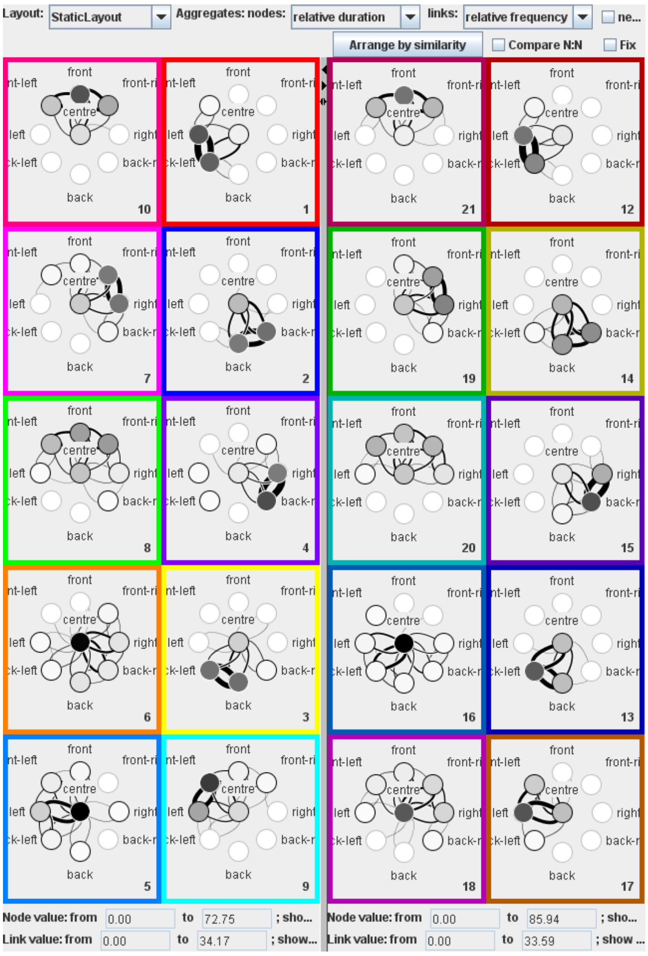

Figure 13. Behaviours of corresponding field players of two teams are visually compared by juxtaposition. The left and right panels represent the players of the home and away teams, respectively.

left, possibly, to counterbalance the striker's shift to the front right. The player number 7 (bright pink frame) shifted from the front-right more to the right, possibly, to give more space and support to the striker (10). Evidently, the home team decided to change its attacking pattern. The defense was slightly adjusted by the player number 3 (yellow, in the upper row of Fig. 14) moving more to the back. The largest change in the guest team was that player number 15 (lower left corner, in a blue frame) reduced his presence on the right and in the centre and spent more time in the back-right position. Instead, player 20 (upper right) took more responsibility for the right and central positions.

This example demonstrates the use of STGs for investigation of individual movement behaviours. It also shows how behaviour changes over time can be explored.

Discussion and conclusion

The examples demonstrate that the representation of movement data in the form of state transition graphs may be useful for analysing the semantics of movement behaviours. This representation is applicable when locations visited by moving objects or combinations of properties of their movements can be semantically interpreted and conceptualised as repeatedly occurring states. A state is a meaningful generalisation of multiple visited locations or multiple movements having something in common, such that the meaning can be expressed by a suitable label. Hence, the STG representation provides a highly abstracted view of movement behaviours, which allows the analyst to focus on general patterns in terms of the frequencies and durations of the states and transitions between them.

We have demonstrated several kinds of possible applications for the STG representation. In the example of VAST Challenge 2014, we studied behaviours of people in terms of their activities, which were performed in specific semantic categories of places. For VAST Challenge 2015, we analysed people's tendencies to visit particular categories of locations. The Milan example refers to applications studying the use of space in terms of general types of spatial locations (e.g., land use categories, street types, etc.) by moving objects depending on the time of movement, trip origin and/or destination, or other characteristics of the movement or the moving objects. The football example shows how STGs can be helpful in exploration of behaviours of moving objects in a group (team).

In our paper, we have described the way of transforming movement data available in the form of trajectories into state transition graphs. Various transformations of movement data have been described previously in the literature (Andrienko et al. (2013b)), but this transformation has not been systematically considered yet. We have explained why the STG representation of movement data is not a replacement but a complement to other representations. Thus, simultaneous use of representations in the form of trajectories and in the form of STGs allows combining spatial, temporal, and semantic analysis of movement. The STG representation alone is insufficient due to its very high level of abstraction from the original physical space where the movement takes place and due to aggregation of time. Furthermore, the STG representation may need to be combined with the state sequence representation when it is important to analyse state sequences and not only direct transitions between states.

Another major contribution of this paper is definition of the properties and behaviour of an STG visualisation tool that are required for the tool to be useful for movement analysis. Taking into account that such a tool needs to be used in combination with other tools working on other representations of movement data, we pay much attention to defining the behaviour of the STG tool required for linking it to other tools. We consider three analytical meta-operations, querying, partitioning, and direct selection, which are, in essence, different approaches to defining and preparing data portions for performing comparisons. These meta-operations are generalisations of many existing and conceivable tools and techniques commonly used in analysing various types of data. We make an abstraction from all possible ways in which these meta-operations can be accomplished and consider only the structure of the obtained result. On the highest level of abstraction, the result is two or more subsets of the data to be examined and compared. When several analytical tools are complementarily used in data analysis, the results of operations performed by means of one of them are propagated to all others. The other tools must take these results into account, to allow integrated analysis. In particular, visual displays must be able to represent the resulting data subsets.

Consequently, we define how an STG visualisation tool presents data subsets resulting from the meta-operations and how it supports comparisons between the subsets. We also define how the tool should deal with results of multiple meta-operations: it should allow the analyst to store, restore, and compare results of different operations. These high-level definitions can be treated as a kind of meta-design, which can be instantiated in a variety of specific designs by choosing particular graph drawing techniques, layout algorithms, user interface elements, modes of user interaction with the display, etc. The specific realisation of this meta-design used in writing this paper should be considered as merely illustrative.

As a final note, it is worth mentioning that the definition (meta-design) of an STG visualisation tool presented in this paper is not specific to movement data. It can also be applied to other kinds of data representable as state sequences, such as event logs, patient state records, time series of attribute values (after segmentation and annotation), and others.

Acknowledgements

This research was supported by EU within projects SoBigData Research Infrastructure (grant agreement 654024) and VaVeL ”Variety, Veracity, VaLue: Handling the Multiplicity of Urban Sensors” (grant agreement 688380), and by German Research Foundation DFG within Priority Research Program 1894 ”Volunteered Geographic Information: Interpretation, Visualisation and Social Computing”.

References

Aggarwal CC and Reddy CK (eds.) (2014) *Data Clustering: Algorithms and Applications*. CRC Press. ISBN 978-1-46-655821-2. URL http://www.crcpress.com/product/isbn/9781466558212.

Aigner W, Miksch S, Schumann H and Tominski C (2011) *Visualization of Time-Oriented Data*. Human-Computer Interaction Series. Springer. ISBN 978-0-85729-078-6.

Andrienko G, Andrienko N, Hurter C, Rinzivillo S and Wrobel S (2013a) Scalable analysis of movement data for extracting and exploring significant places. *Visualization and Computer Graphics, IEEE Transactions on* 19(7): 1078–1094. DOI: 10.1109/TVCG.2012.311.

Andrienko GL, Andrienko NV, Bak P, Keim DA and Wrobel S (2013b) *Visual Analytics of Movement*. Springer. ISBN 978-3-642-37582-8. DOI:10.1007/978-3-642-37583-5. URL http://dx.doi.org/10.1007/978-3-642-37583-5.

Andrienko N, Andrienko G, Barrett L, Dostie M and Henzi P (2013c) Space transformation for understanding group movement. *Visualization and Computer Graphics, IEEE Transactions on* 19(12): 2169–2178. DOI:10.1109/TVCG.2013.193.

Andrienko N, Andrienko G and Fuchs G (2013d) Towards privacy-preserving semantic mobility analysis. In: Pohl M and Schumann H (eds.) *EuroVis Workshop on Visual Analytics*. The Eurographics Association. ISBN 978-3-905674-55-2. DOI: 10.2312/PE.EuroVAST.EuroVA13.019-023.

Andrienko N, Andrienko G and Fuchs G (2014) Analysis of mobility behaviors in geographic and semantic spaces. In: *Visual Analytics Science and Technology (VAST), 2014 IEEE Conference on*. pp. 341–342. DOI:10.1109/VAST.2014.7042556.

Andrienko N, Andrienko G, Fuchs G and Jankowski P (2015) Scalable and privacy-respectful interactive discovery of place semantics from human mobility traces. *Information Visualization* DOI:10.1177/1473871615581216.

Andrienko N, Andrienko G, Stange H, Liebig T and Hecker D (2012) Visual analytics for understanding spatial situations from episodic movement data. *KI - Künstliche Intelligenz* 26(3): 241–251. DOI: 10.1007/s13218-012-0177-4. URL http://dx.doi.org/10.1007/s13218-012-0177-4.

Ankerst M, Breunig MM, Kriegel HP and Sander J (1999) Optics: ordering points to identify the clustering structure. In: *ACM Sigmod Record*, volume 28. ACM, pp. 49–60.

Archambault D, Abello J, Kennedy J, Kobourov S, Ma KL, Miksch S, Muelder C and Telea A (2014) Temporal multivariate networks. In: Kerren A, Purchase H and Ward M (eds.) *Multivariate Network Visualization*, *Lecture Notes in Computer Science*, volume 8380. Springer International Publishing. ISBN 978-3-319-06792-6, pp. 151–174. DOI:10.1007/978-3-319-06793-3_8.

Beck F, Burch M, Diehl S and Weiskopf D (2014) The State of the Art in Visualizing Dynamic Graphs. In: Borgo R, Maciejewski R and Viola I (eds.) *EuroVis - STARs*. The Eurographics Association. ISBN -. DOI:10.2312/eurovisstar.20141174.

Becker RA and Cleveland WS (1987) Brushing scatterplots. *Technometrics* 29(2): 127–142.

Blaas J, Botha C, Grundy E, Jones M, Laramee R and Post F (2009) Smooth graphs for visual exploration of higher-order state transitions. *Visualization and Computer Graphics, IEEE Transactions on* 15(6): 969–976.

Bouvier DJ and Oates B (2008) Evacuation traces mini challenge award: Innovative trace visualization staining for information discovery. In: *IEEE VAST*. IEEE Computer Society. ISBN 978-1-4244-2935-6, pp. 219–220. URL http://dblp.uni-trier.de/db/conf/ieeevast/ieeevast2008.html#Bouvier08.

Ester M, Kriegel HP, Sander J, Xu X et al. (1996) A density-based algorithm for discovering clusters in large spatial databases with noise. In: *Kdd*, volume 96. pp. 226–231.

Gibson H, Faith J and Vickers P (2013) A survey of two-dimensional graph layout techniques for information visualisation. *Information Visualization* 12(3-4): 324–357. DOI:10.1177/1473871612455749.

Gleicher M, Albers D, Walker R, Jusufi I, Hansen CD and Roberts JC (2011) Visual comparison for information visualization. *Information Visualization* 10(4): 289–309. DOI:10.1177/1473871611416549.

Hadlak S, Schumann H and Schulz HJ (2015) A Survey of Multi-faceted Graph Visualization. In: Borgo R, Ganovelli F and Viola I (eds.) *Eurographics Conference on Visualization (EuroVis) - STARs*. The Eurographics Association. DOI:10.2312/eurovisstar.20151109.

Martin A and Ward M (1995) High dimensional brushing for interactive exploration of multivariate data. In: *Visualization, 1995. Visualization '95. Proceedings., IEEE Conference on*. pp. 271–. DOI:10.1109/VISUAL.1995.485139.

Monroe M, Lan R, Lee H, Plaisant C and Shneiderman B (2013) Temporal event sequence simplification. *IEEE Trans. Vis. Comput. Graph.* 19(12): 2227–2236. DOI:10.1109/TVCG.2013.200.

Norris J (1998) *Markov Chains*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press. ISBN 9781107393479.

Plaisant C, Milash B, Rose A, Widoff S and Shneiderman B (1996) Lifelines: Visualizing personal histories. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '96. New York, NY, USA: ACM. ISBN 0-89791-777-4, pp. 221–227. DOI:10.1145/238386.238493.

Pretorius AJ and van Wijk JJ (2006) Visual analysis of multivariate state transition graphs. *IEEE Trans. Vis. Comput. Graph.* 12(5): 685–692. DOI:10.1109/TVCG.2006.192.

Riehmann P, Hanfler M and Froehlich B (2005) Interactive sankey diagrams. In: *Information Visualization, 2005. INFOVIS 2005. IEEE Symposium on*. IEEE, pp. 233–240.

Roberts JC (2007) State of the art: Coordinated & multiple views in exploratory visualization. In: *Proceedings of the Fifth International Conference on Coordinated and Multiple Views in Exploratory Visualization*, CMV '07. Washington, DC, USA: IEEE Computer Society. ISBN 0-7695-2903-8, pp. 61–71. DOI:10.1109/CMV.2007.20. URL http://dx.doi.org/10.1109/CMV.2007.20.

Shneiderman B (1994) Dynamic queries for visual information seeking. *IEEE Software* 11(6): 70–77. DOI:10.1109/52.329404. URL http://dx.doi.org/10.1109/52.329404.

Spence R and Tweedie L (1998) The attribute explorer: information synthesis via exploration. *Interacting with Computers* 11: 137–146.

Tufte ER (1983) *The Visual Display of Quantitative Information*. Cheshire, CT, USA: Graphics Press. ISBN 0-9613921-0-X.

van Ham F, van de Wetering H and van Wijk JJ (2001) Visualization of state transition graphs. In: Andrews K, Roth SF and Wong PC (eds.) *IEEE Symposium on Information Visualization 2001 (INFOVIS'01), San Diego, CA, USA, October 22-23, 2001*. IEEE Computer Society. ISBN 0-7695-1342-5, pp. 59–63. DOI:10.1109/INFVIS.2001.963281.

van Ham F, van de Wetering H and van Wijk JJ (2002) Interactive visualization of state transition systems. *IEEE Trans. Vis. Comput. Graph.* 8(4): 319–329. DOI:10.1109/TVCG.2002.1044518.

VC2014 (2014) Vast challenge 2014. URL http://www.vacommunity.org/VAST+Challenge+2014.

VC2015 (2015) Vast challenge 2015. URL http://vacommunity.org/VAST+Challenge+2015.

von Landesberger T, Bremm S, Andrienko NV, Andrienko GL and Tekusova M (2012) Visual analytics methods for categoric spatio-temporal data. In: *2012 IEEE Conference on Visual Analytics Science and Technology, VAST 2012, Seattle, WA, USA, October 14-19, 2012*. IEEE Computer Society. ISBN 978-1-4673-4752-5, pp. 183–192. DOI:10.1109/VAST.2012.6400553.

von Landesberger T, Brodkorb F, Roskosch P, Andrienko NV, Andrienko GL and Kerren A (2016) Mobilitygraphs: Visual analysis of mass mobility dynamics via spatio-temporal graphs and clustering. *IEEE Trans. Vis. Comput. Graph.* 22(1): 11–20. DOI:10.1109/TVCG.2015.2468111.

Vrotsou K, Forsell C and Cooper M (2010) 2d and 3d representations for feature recognition in time geographical diary data. *Information Visualization* 9(4): 263–276. DOI:10.1057/ivs.2009.30.

Vrotsou K, Johansson J and Cooper MD (2009) Activitree: Interactive visual exploration of sequences in event-based data using graph similarity. *IEEE Trans. Vis. Comput. Graph.* 15(6): 945–952. DOI:10.1109/TVCG.2009.117.

Wongsuphasawat K, Guerra Gómez JA, Plaisant C, Wang TD, Taieb-Maimon M and Shneiderman B (2011) Lifeflow: Visualizing an overview of event sequences. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '11. New York, NY, USA: ACM. ISBN 978-1-4503-0228-9, pp. 1747–1756. DOI:10.1145/1978942.1979196.

Xu C, Ji M, Chen W and Zhang Z (2010) Identifying travel mode from gps trajectories through fuzzy pattern recognition. In: *Fuzzy Systems and Knowledge Discovery (FSKD), 2010 Seventh International Conference on*, volume 2. pp. 889–893. DOI:10.1109/FSKD.2010.5569105.

Zhao J, Forer P and Harvey AS (2008) Activities, ringmaps and geovisualization of large human movement fields. *Information Visualization* 7(3): 198–209. DOI:10.1057/palgrave.ivs.9500184.

Zhao J, Liu Z, Dontcheva M, Hertzmann A and Wilson A (2015) Matrixwave: Visual comparison of event sequence data. In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15. New York, NY, USA: ACM. ISBN 978-1-4503-3145-6, pp. 259–268. DOI:10.1145/2702123.2702419.