

STRUCTURAL COMPONENTS IN MULTIADJUSTABLE ROAD TRAFFIC MODELS: THEIR ROLE AND THE MEANS OF GENERATING THEIR TOPOLOGY

Petr Zelenka and Jana Hájková
Department of Computer Science and Engineering
University of West Bohemia
Univerzitní 8, 306 14 Plzeň, Czech Republic
Email: {pzeli,hajkovaj}@kiv.zcu.cz

KEYWORDS

Road traffic modeling and simulation, multiadjustable model, structural component.

ABSTRACT

This article presents advanced features of structural components in multiadjustable road traffic models. It shows that by making the structural components the active elements of the model, it is not only possible to enable multiple-level-of-detail modeling of traffic, but it also makes solving some corner-case situations much easier. Further, it deals with the problem of describing the topology of the traffic network that could be used for both macroscopic and microscopic structural components. It identifies a minimum set of information for such description and provides methods for deriving the remaining information from this minimum set.

INTRODUCTION

Modeling and simulation of road traffic has gained in popularity in recent years. This is not surprising, because it constitutes the only tool available so far for predicting behavior of a traffic system. Using traffic models, it is possible, for example, to evaluate the impact of the traffic restrictions related to a road work (whether construction or maintenance) and to determine which combinations of road works can be performed simultaneously without causing unnecessary congestion, or to compare the performance of several different road design alternatives. In connection with the recent onset of intelligent traffic management and information systems, traffic models can also be used to test these systems during their development in a real-like environment.

In our previous research (Zelenka, 2009), we proposed a general framework and a methodology for refactoring the structure of road traffic models to agility. We found that by introducing structural and behavioral components (see also the next section), it is possible to make use of component substitutability for easy changes in both structure and behavior of the models. Further, by making the structural components the active elements of the model and introducing component adapters, it is possible to simulate the traffic at different level of details in different parts of

the model. Together, these two concepts lead to what we started to call multiadjustable models, because the features of such a model can be changed almost arbitrarily.

THE ROLE OF STRUCTURAL COMPONENTS

Today, there are basically two different approaches to road traffic modeling. The first one, called macroscopic, describes the traffic as a whole in terms of physical quantities such as traffic flow and traffic density. The state of such models is changed based on differential (or, in the case of computerized models, difference) equations describing fluid dynamics. The second approach, called microscopic, describes the traffic at a greater level of detail, namely in terms of individual traffic participants (e.g., cars, trucks, streetcars, or even pedestrians). The state of such models is changed based on complex algorithms describing the state variables (e.g., position, speed, or acceleration) of each traffic participant. More details can be found, for example, in (Hämäläinen, 2006).

Regardless of which approach is used, there is an aspect that all traffic models have in common. It is the underlying streets and intersections, called a traffic network in short. In multiadjustable road traffic models, the traffic network is composed of a set of interconnected structural components. Originally, we defined a structural component as either an intersection or a road segment connecting two intersections, but smaller parts of traffic network can also be considered structural components, depending on the situation. For instance, a road segment can be subdivided into two or more structural components at the points where the number of traffic lanes changes (see Figure 1). Similarly, modeling individual traffic lanes by separate structural components can also be helpful under certain conditions.

In order to enable multiple-level-of-details modeling, the overall model needs to be subdivided into several areas, each modeled at either macroscopic or microscopic level of detail. For this purpose, structural components are a good choice, because they can easily provide different granularity of this subdivision. So, each structural component shall be looked upon as a standalone submodel of a specific small area.

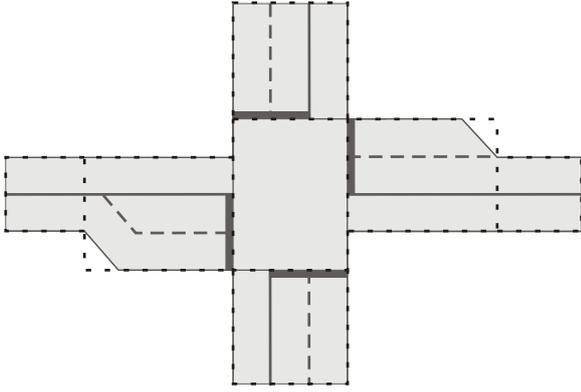


Figure 1: Example of decomposing a part of a traffic network into structural components (bounded by dashed lines). In this case, any change in number of traffic lanes constitutes a structural components boundary.

This also means that we need to distinguish between macroscopic and microscopic structural components, because their modes of operation are different. Macroscopic structural components obtain the values of the physical quantities describing the traffic dynamics from their neighbors, compute new values of these quantities based on the corresponding difference equations, and pass these new values to the neighbors again. Their behavior is therefore rather simple and constant. On the other hand, the behavior of microscopic structural components is much more complex. They need to accept traffic participants coming from structural components adjacent to them, then to pass these traffic participants along their topology using some complex algorithms and finally to pass them to some other adjacent structural components. Their behavior is composed of many submodels (e.g., car-following model or lane-changing model), so they can make use of behavioral components in order to make their behavior adjustable to the conditions being modeled, such as local traffic law. For more details about behavioral components, see (Zelenka, 2009).

Because macroscopic and microscopic structural components communicate in different terms (physical quantities vs. traffic participants), they cannot be interconnected directly. Instead, adapters (i.e., intermediate components that serve as interpreters) must be placed between them. In the direction from a microscopic to a macroscopic component, the adapter needs to obtain the values of the respective quantities using statistical analysis, in the opposite direction, it needs to generate the traffic participants according to the values of these quantities.

The preceding paragraphs imply that structural components are the active elements of the simulation. That is, they are responsible for updating the state and position of the traffic participants. This is a big difference compared

to most existing microscopic road traffic simulation tools, where the traffic participants update their state and position themselves. However, making the structural components active has some additional advantages. For example, consider the situation from Figure 2. There are four cars situated at an uncontrolled intersection. At uncontrolled intersections, cars are required to give right of way to the cars approaching from the right side in most countries. Consequently, each of these cars is required to give right of way, effectively causing a deadlock. To resolve this situation, the drivers need to agree on who will travel through the intersection first. When the cars are the active elements, it is not easy to detect this situation, nor it is to resolve it. In fact, it requires some distributed algorithms to be implemented within the model. When we make the structural components to care for updating car states and positions, detecting and resolving this situation is easy, because there is only one element (the structural component) responsible for doing it.

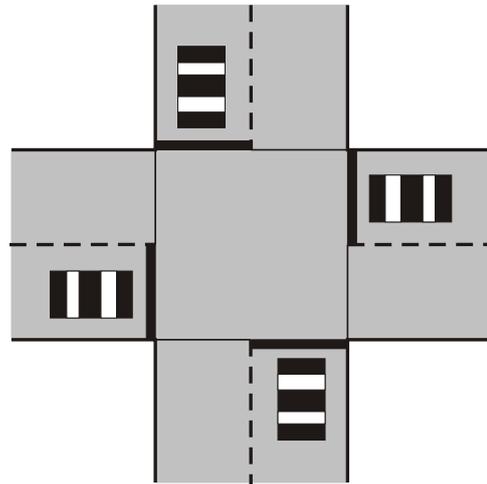


Figure 2: A deadlock situation caused by four cars situated at an uncontrolled intersection.

Moreover, even if the cars were the active elements, some structural components would also have to implement some behavior. An example can be traffic-light operated intersections, which need to implement some logic determining which traffic participants are allowed to travel through it at which times.

DESCRIBING AND GENERATING THE TOPOLOGY OF STRUCTURAL COMPONENTS

Most of the existing microscopic road traffic models are based on cellular automata. According to (Chopard and Droz, 1998), a cellular automaton is a lattice of cells in d -dimensional space, where each cell is associated with a Boolean flag representing the state of the cell (i.e., empty or occupied), supplemented by a rule (same for all the cells) describing the time evolution of these

states. In the case of road traffic models, $d = 2$ and the state-changing rule is represented by the complex behavioral model mentioned in the preceding section. One of the most well-known (and still very popular, especially for its simplicity) road traffic models based on cellular automata is the Nagel-Schreckenberg model; see (Nagel and Schreckenberg, 1992).

For cellular-automata-based microscopic models, the topology of structural components will be composed of individual cells. Unfortunately, macroscopic structural components are spatially compact. This difference is somewhat unpleasant, because it means that we either need to maintain two separate descriptions of the traffic network or need to find a description that can be used for both macroscopic and microscopic models. Further, from our previous experience we know that storing information about a traffic network at the microscopic level of detail shall be avoided, because it constitutes a serious performance bottleneck at simulation startup when an enormous amount of data needs to be loaded from some slow external memory. To overcome these two issues, we need to find a minimum set of information that unambiguously describes the traffic network and from which the rest of information (e.g., the positions of the individual cells) can be generated.

In general, there are two basic requirements that the topology of structural components shall satisfy. The first one is that lengths are reproduced to some scale. Satisfying this requirement is essential, because any disproportion between lengths in the model and lengths in the real situation inevitably means an error in results of the simulation. The second one is that the overall topology of the traffic network in the model resembles its real counterpart as closely as possible. This requirement is in fact completely unnecessary for the correctness of the model, but is important for proper visualization of the simulation results. Since a long time ago, people have been accustomed to visualize geospatial information in the form of maps. So providing the results of the simulation in a map-like user interface makes them easier to understand than, for example, writing them as an obscure matrix of numbers.

There are two aspects in describing the topology of a structural component. The first one is the general topology (i.e., the shape and dimensions) and the position in space. In this case, we can follow geographic information systems, which also deal with this aspect. The second one, specific to the domain of cellular-automata-based simulation models, is the layout of the cells inside this general topology.

Describing Traffic Lanes and Road Segments

Let's look at the structural components in greater detail. The most simple structural component to think about is

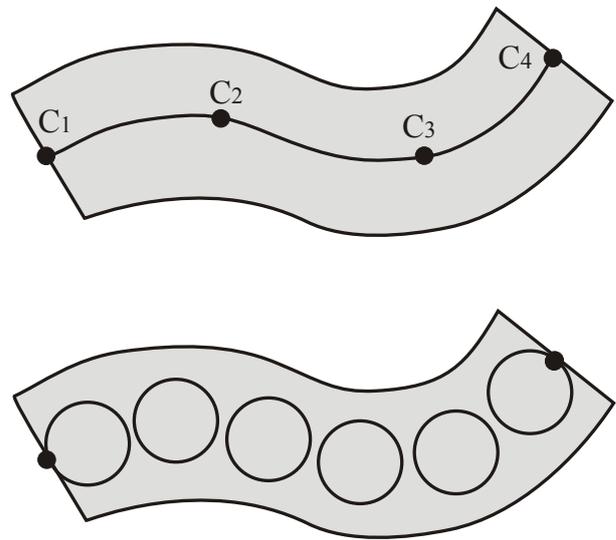


Figure 3: An example of reducing the representation to a guideline and further to a sequence of control points (C_1, C_2, C_3, C_4) through which it passes (top). An example of generating cells along the guideline (bottom).

probably a traffic lane. Any road segment can be composed of one or more traffic lanes. One traffic lane constitutes an one-way road. Where more traffic lanes are present within a road segment, the road can be either an one-way road (all traffic lanes oriented in the same direction) or a two-way road (some of the traffic lanes oriented in one direction, some in the opposite one). Describing the general topology of a single traffic lane is actually quite easy. If we assume that the traffic lane has a constant width (most often, the minimum width defined by some construction standard is used in real situations), we can reduce the description to a single curve (see Figure 3, top). Further, we can represent a curve by a set of points through which it passes, because the original curve can be regained by interpolating them. If we had many points, we could simply connect them by straight lines. In our case, when we are trying to find a minimum set of these points, we more likely want to use some interpolating curve. As a conclusion, traffic lanes can be described by a set of node points.

Describing Intersections

With intersections, the situation is more complicated. The general topology of an intersection can be derived from the traffic lanes that are connected by the intersection or, to be more precise, their corresponding end points. In fact, the tangential vectors to the traffic lanes in these end points are also needed (see Figure 4), but they can be easily calculated using the waypoints adjacent to the end points. Therefore, the minimum set of information to describe the general topology of an intersection is the set of end points of the adjacent traffic lanes. However, to generate the cells within the intersection, some additional information is necessary.

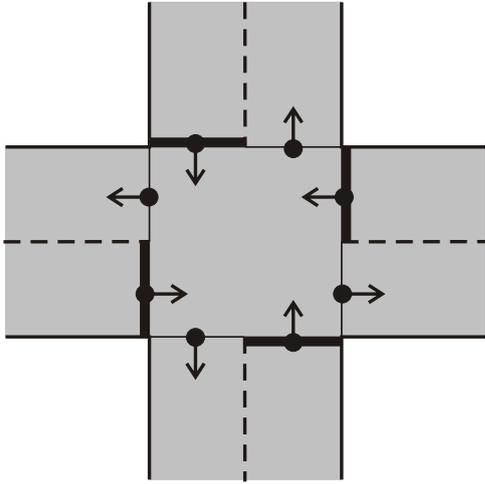


Figure 4: An example of representing an intersection by the end points of the adjacent road segments and the tangential vectors to these road segments in their end points.

We need to know which pairs of traffic lanes shall be interconnected (i.e., to which outgoing traffic lanes it is possible to travel from each ingoing traffic lane). Based on this information, we can generate a guideline representing the trajectory through the intersection for each such pair. We have studied the trajectories of vehicles traveling through intersections and we came to a conclusion that these trajectories are best represented by rational Bézier curves.

A rational Bézier curve is a special Bézier curve that assigns each control point an adjustable weight in order to become a closer approximation to an arbitrary shape. A rational Bézier curve of degree n (i.e., defined by $n + 1$ control points) can be expressed by Equation (1), where P_i is the i -th control point, w_i is its weight, and the parameter t takes the values from 0 to 1.

$$Q(t) = \frac{\sum_{i=0}^n \binom{n}{i} t^i (1-t)^{n-i} P_i w_i}{\sum_{i=0}^n \binom{n}{i} t^i (1-t)^{n-i} w_i} \quad (1)$$

For our purpose, a rational Bézier curve of degree 2 is appropriate. The control points (see Figure 5, top) are the two end points of the interconnected road segments (S, E) and either the point lying in the middle of the straight line connecting these two end points (if they are collinear) or the intersection point (C) of the tangential vectors to the interconnected road segments (in all other cases). The weights w_i are equal to 1.0 in points S and E and greater than 1 in point C . In our experiments, a value around 2.5 provided the most realistic trajectory.

Generating Individual Cells

So far, we dealt with the general topology of structural components. But, as mentioned above, we also need

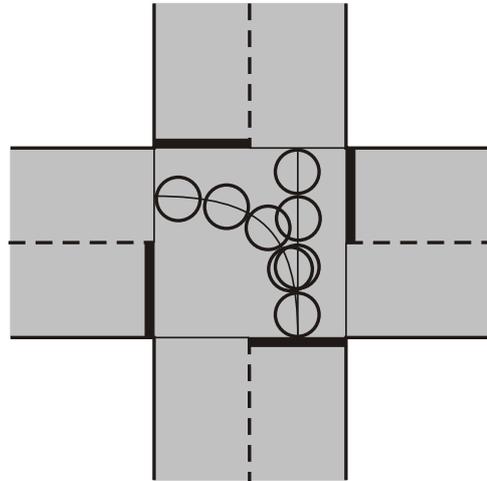
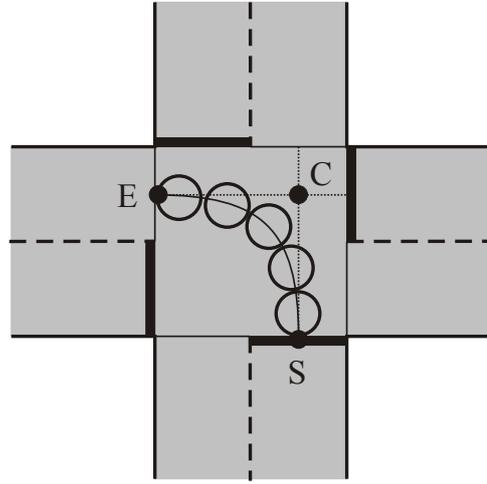


Figure 5: An example of a trajectory through an intersection defined by a rational Bézier curve (top). An example of overlapping cells generated on different guidelines (bottom).

to deal with the other aspect of the structural components, that is generating the individual cells. In most cellular-automata-based road traffic models, circular cells are used. The center of each cell shall lie on a trajectory which the traffic participants follow, so we can use the above-described curves as guidelines for the cell generation (see Figure 3, top vs. bottom). The diameter of cells then uniquely defines the number of cells that are to be generated on a guideline with a certain length. In practice, these diameters differ from model to model. The reason is that their value, in connection with the length of the time step in which the state of the model is updated, determines the possible discrete values of speed that is possible to model. Most often, diameters range from 2 to 3 meters.

Although the centers of the individual cells are uniquely defined by the length of the guideline and the diameter of the cells, it is not so straightforward to calculate them.

The first problem is that the length of the guideline needs to be determined. For most curves, there is unfortunately no formula to return the length of it. And, even if there is one, it requires the curve to be defined analytically, because it usually involves integration. So, we need to use another approach. For curves that are defined using a parameter, it is possible to generate points lying on the curve by substituting in some increasing values for the parameter and then to approximate the curve by a polyline (i.e., a set of straight lines connecting adjacent points). This is sometimes called tessellation of the curve. Calculating the length of a polyline is easy, so the main concern is to find a reasonable number of points to use, because too few points means an inaccurate approximation and too many points means an unnecessary computational overhead. The second problem is that the tessellation needs to be equidistant so that the cells fill the space uniformly. This may seem easy at first glance, but most curves (rational Bézier curves not being an exception) cannot be tessellated equidistantly by increasing their parameter by a constant step. Instead, a technique called reparametrization by arc length needs to be used; see (Schneider and Eberly, 2002, p. 890). Reparametrization by arc length is based on solving Equation (2) for some value x , where $Q(x)$ is a point on the curve for parameter value x , $Q(S)$ is the starting point of the curve, and L is the desired length from this starting point.

$$Q(x) - Q(S) = L \quad (2)$$

Typically, Newton's method is used, but this also causes an unnecessary computational overhead. In our case, performing the tessellation for some smaller parameter step and saving the resulting points together with their distance from the starting point of the curve into a search table seems to be a better solution, as it allows us later to use the nearest point from this table.

Now, we are able to generate the cells. However, there is still one problem to be solved. In intersections, there are guidelines that either intersect in some point or are leading close to each other in some area. Consequently, generating the cells on these guidelines leads to cells that overlap (see Figure 5, bottom). This means that, in the real world, there is not enough space for two cars to fit in the area modeled by these cells. There are basically two different solutions. The first one is to replace these cells by a single one (its center could be the midpoint of the line joining the centers of the replaced cells, for example). This would decrease the number of cells in the model, but would also cause unrealistic deviations in car motions during visualization. The second solution is to use a single state (see the Chopard and Droz definition of a cellular automaton above) for all the overlapping cells (i.e., to mutually exclude the cars from entering the area).

Referencing to a Particular Position in a Traffic Lane

Finally, we need to provide a means for associating additional information with parts of the traffic network. Examples of such information may be speed restriction or reserving a traffic lane for some particular traffic only. Because this information can begin and end somewhere inside the structural components, we need to associate it with the individual cells to which it relates. This means that we need to reference cells which do not exist yet. We can describe the additional information by its type and its impact point (i.e., a point where its validity begins or ends). The impact point can be determined using the distance from the starting point of the guideline associated with the structural component (see Figure 6). Based on this information, the cell to which this point belongs can be determined.

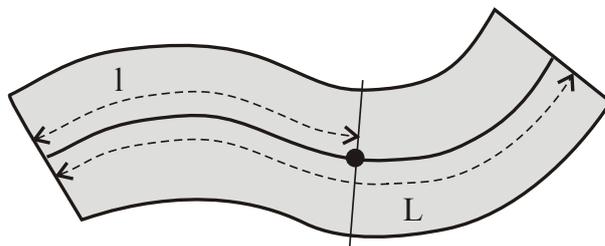


Figure 6: An example of referencing to an impact point based on lengths. Here, the impact point is located in l/L of the structural component.

RESULTS

The vehicle trajectories represented by a polyline generated using rational Bézier curves (weights in end points $w_{0,2} = 1.0$, weight in the middle control point $w_1 = 2.5$) and the equidistantly distributed cells generated along this trajectories are presented in Figures 7 to 10. It can be seen that the shapes of the trajectories clearly resemble the shapes of trajectories of vehicles driving through a smooth bend. The figures showing overlapping cells can seem confused at first sight, but it should be kept in mind that the cells only serve the purpose of computing the simulation state and are typically not visualized in any way. Conversely, their placement along the smooth vehicle trajectories make the visualization of cars traveling through an intersection more natural and therefore pleasurable.

SUMMARY

In this paper, we dealt with structural components in multiaadjustable road traffic models. In the first part, we presented some reasons for making them the active part of the model and showed that it has more advantages than just enabling multiple-level-of-detail modeling. In the second part, we investigated the possibility of finding a single description of their topology suitable for both macroscopic and microscopic modeling. We

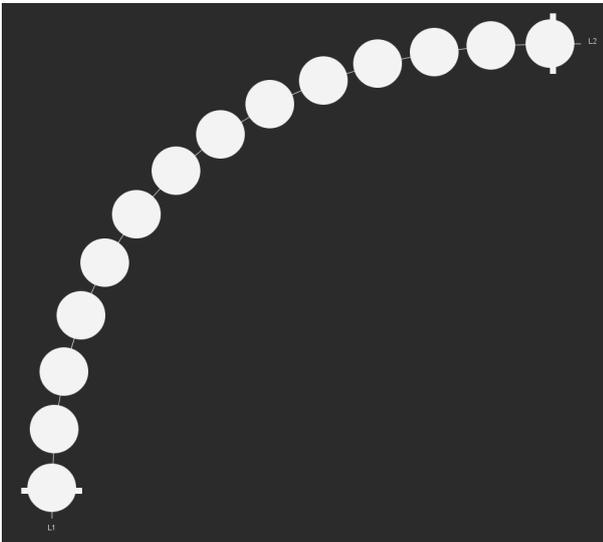
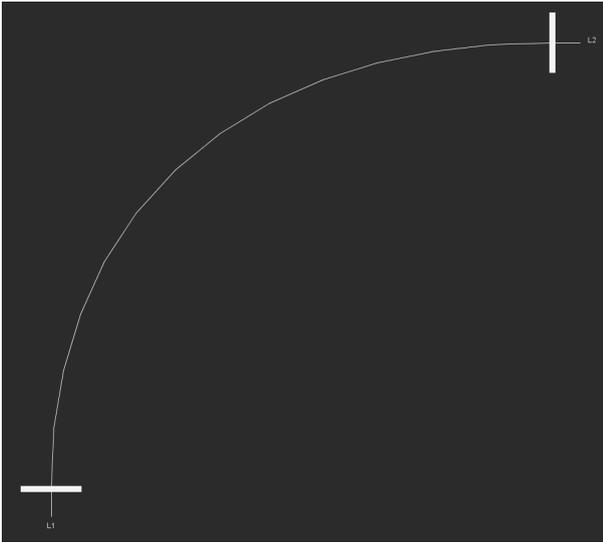


Figure 7: An example of a trajectory through an intersection defined by a rational Bézier curve (top). An example of cells generated along this trajectory (bottom).

identified a minimum set of information that is necessary to describe a traffic network and provided some methods for deriving the rest of information from this minimum set.

The experience gained from this research and the previous research of behavioral components will be used to develop an experimental multiadjustable road traffic model. Together with the experience from (Hájková, 2005), it will be also used to support this experimental model in the areas of preparing the maps of traffic networks and visualizing the simulation results.

ACKNOWLEDGMENT

This work was partially supported by the Grant Agency of the Czech Republic (project number 201/08/0266).

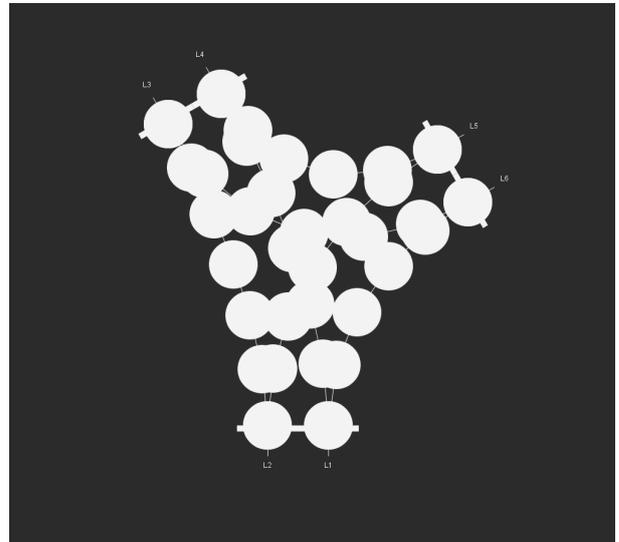
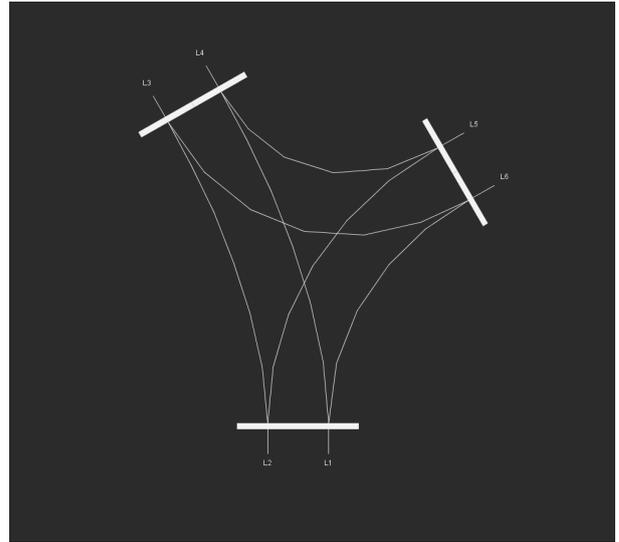


Figure 8: An example of trajectories through a Y-shaped intersection (top). An example of overlapping cells generated along these trajectories (bottom).

REFERENCES

- Chopard, B. and Droz, M. (1998). *Cellular Automata Modeling of Physical Systems*. Cambridge University Press, Cambridge, UK.
- Hájková, J. (2005). *Grafický editor schémat dopravních systémů*. MS thesis, University of West Bohemia, Plzeň, Czech Republic.
- Hämäläinen, A. (2006). *Studies of Traffic Situations Using Cellular Automata*. PhD dissertation, Helsinki University of Technology, Espoo, Finland.
- Nagel, K. and Schreckenberg, M. (1992). A Cellular Automaton Model for Freeway Traffic. In *Journal de Physique I*, Vol. 2, pp. 2221–2229.

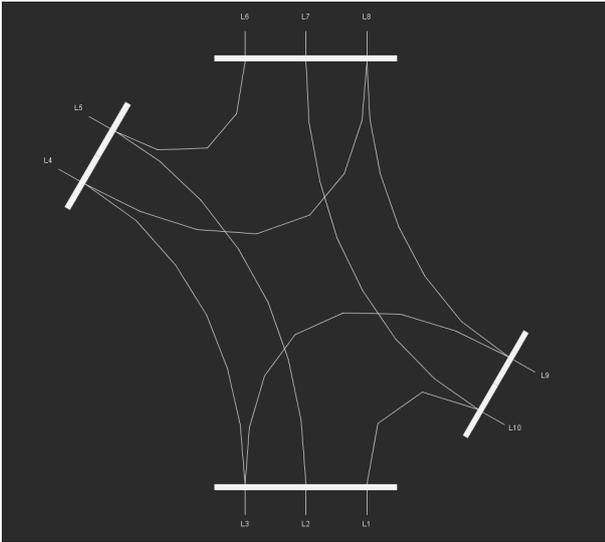


Figure 9: An example of trajectories through an X-shaped intersection (top; some trajectories have been omitted for the sake of clarity). An example of overlapping cells generated along these trajectories (bottom).

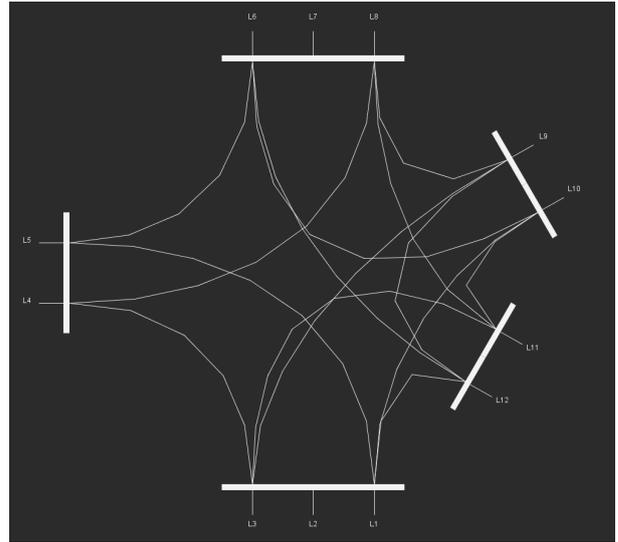


Figure 10: An example of trajectories through an irregularly shaped intersection (top; some trajectories have been omitted for the sake of clarity). An example of overlapping cells generated along these trajectories (bottom).

Schneider, P. and Eberly, D.H. (2002). *Geometric Tools for Computer Graphics*. Morgan Kaufman, San Francisco, CA, USA.

Zelenka, P. (2009). Introducing Software Components to Road Traffic Modeling and Simulation. In *Proceedings of the 6th Vienna Conference on Mathematical Modelling (MATHMOD 09)* [CD-ROM]. ARGESIM, Vienna, Austria.

AUTHOR BIOGRAPHIES

PETR ZELENKA is a PhD student of computer science at the University of West Bohemia. His research interests include software architectures, software development practices and computer simulation. He received a BS and a MS in computer science from the University of West Bohemia in 2005 and 2007, respectively. Contact him at pzeli@kiv.zcu.cz.

JANA HÁJKOVÁ is a PhD candidate and teaching assistant of computer science at the University of West Bohemia. Her research interests include data visualization and computer simulation. She received a MS in computer science from the University of West Bohemia in 2005. Contact her at hajkovaaj@kiv.zcu.cz.