Abstract. Dense human flow has been a concern for the safety of public events for a long time. Macroscopic pedestrian models, which are mainly based on fluid dynamics, are often used to simulate huge crowds due to their low computational costs (COLOMBO and ROSINI, 2005). Similar approaches are used in the field of traffic simulations (LIGHTHILL and WHITHAM, 1955). A combined macroscopic simulation of vehicles and pedestrians is extremely helpful for all-encompassing traffic control. Therefore, we developed a hybrid model that contains networks for vehicular traffic and human flow. This comprehensive model supports concurrent multi-modal simulations of traffic and pedestrians (BIEDERMANN, KIELAR, AUMANN, LAI and OSORIO, 2015).

1 Introduction

1.1 Hartmann-Sivers Model

Every year, about 2000 people die due to the movement of large human crowds (HUGHES, 2003). The growing eventisation of social and public life in the 21st century (HITZLER, 2010) increases the probability and frequency of organized events attended by thousands or millions of people. In the past, fatal accidents have occurred repeatedly at such large events. Examples include the catastrophe in the Heysel stadium in 1985 (LEWIS, 1989) with 39 fatalities or the Loveparade disaster in 2010 with 21 deaths (HITZLER et al., 2011). Professional crowd control increases the safety of public events and lowers the probability of fatal accidents in the context of large human flows. Pedestrian dynamics simulations support successful crowd management by helping organizers of public events to foresee and prevent dangerous situations. Macroscopic pedestrian dynamic models like the Hartmann & Sivers mode (HARTMANN and VON SIVERS, 2013) are most suitable for simulating large numbers of people (BIEDERMANN, KIELAR and HANDEL, 2014). The same macroscopic model can be applied to simulate the motion of cars. By combining human and vehicular flow into one model, we can achieve an all-embracing simulation model, which covers crowd and traffic control at the same time (BIEDERMANN, KIELAR, AUMANN, LAI and OSORIO, 2015). Due to the macroscopic nature of these simulation models, we can simulate very large and dense scenarios faster than real time.
1.2 Current Hybrid Modeling

Two different kinds of hybrid modeling exist in the field of traffic and crowd control. The first type combines traffic or pedestrian dynamics models at different scales in order to reduce the overall computational effort. This means that simulations in the hazardous areas of the scenario (places with high densities or small bottlenecks) are calculated with high detail but computationally costly models. At the same time, less hazardous areas of the scenario can be simulated by less detailed but more cost-effective models. This provides us precious results in the interesting parts of the scenario with a reduced total computational cost. TOLBA et al., 2005 as well as MCCREA and MOUTARI, 2010 developed such hybrid models in the field of vehicular flow. Similar models were developed for pedestrian dynamics. A wide range of hybrid modeling exists; from the coupling of very specific pedestrian simulation models (e.g. NGUYEN et al., 2012) to generic coupling frameworks (BIEDELMANN, KIELAR, HANDEL and BORRMANN, 2014). An overview of the current state of the art can be found by IJAZ et al., 2015.

The second type of hybrid modeling does not combine different scales, but different kinds of simulations. GALEA et al., 2008 describe the coupling of pedestrian dynamics and fire simulations and GÖTTLICH et al., 2011 combine an evacuation simulation with the spread of hazardous gases. Our approach is the connection of vehicular traffic with human flow. Some studies have already been carried out in this field. PRETTO et al. (2011) coupled agent-based representations of pedestrians and cars. A macroscopic approach was developed by BORSCHKE et al., 2014. They used the classic Lighthill-Whitham-Richards model (LIGHTHILL and WHITHAM, 1955; RICHARDS, 1956) for the vehicular flow and the model from HUGHES, 2002 for the macroscopic simulation of pedestrians. Their work is important progress in the field of hybrid macroscopic modeling. However, they considered vehicles and humans as two interacting, but separate flows. In reality, cars and pedestrians are not completely separate: drivers exit their cars after finishing driving and become pedestrians and pedestrians may enter their vehicles to become drivers. Our approach is a hybrid model, which is capable of converting humans and vehicles into each other to create a realistic simulation model.

2 The Hybrid Simulation Model CarPed

2.1 The Hybrid approach

It is important to use a single network that combines pedestrian and traffic simulations. Therefore, we introduce an interface which links human and vehicular flow. The movement behavior of cars and pedestrians is based on the macroscopic Hartmann and Sivers model (HARTMANN and VON SIVERS, 2013) and is calculated on a discrete network. Roads for cars and walking paths for pedestrians are represented by the edges of this network. Its nodes represent the links between the different edges and can be considered crossings. Specifically, the nodes have different flow rates. There are special nodes, which represent parking lots, and serve as a connection between roads and walkways. Visitors of the event enter and leave their cars there. To put it another way, the simulated subjects get transformed from vehicles into pedestrians and vice versa. Therefore, the parking lots serve as a connection of the traffic and pedestrian networks. In Figure 1 an exemplary network is
shown. The darker nodes and edges represent streets and their respective intersections, whereas the pedestrians walkways and intersections are represented in a lighter gray. The transformation of cars and pedestrians is carried out at special parking lot nodes. Green numbered entry nodes represent sources, which are the starting points of the visitors (e.g. their hometown). The red marked exit node is the final destination of the visitors (e.g. a large public event), where they are deleted from the network.

Figure 1: An Exemplary Network Consisting of Nodes and Edges

We implemented the Hartmann and Sivers model for pedestrian simulation and extended it to traffic simulation by using different model parameters. The extension was done by applying different maximum capacities to the edges and nodes, using different free flow velocities and other model constants for the simulated subjects. We used data obtained from [WEIDMANN 1993] for the pedestrians and data from [WACHS 2000] for the vehicle simulation. Additionally, we conducted a large field study to receive sufficient data for the transformation process (see Section 3).

2.2 The Hartmann-and-Sivers Model

The Hartmann and Sivers model adopts a structured continuum model on a macroscopic level. The approach basically relies on fundamental diagrams - the relation between fluxes and local densities - as well as the explicit consideration of individual velocities; thus showing better agreement with microscopic pedestrian models ([HARTMANN and VON SIVERS 2013]). The free flow velocity \( v_{ff} \) is an individual property of each pedestrian / vehicle, i.e. a macroscopic, time independent state variable attached to each individual. The free flow velocity describes the highest velocity an individual can reach under optimal circumstances. This property is distributed normally among pedestrians or cars. Thus for modelling human or vehicular flow, the following equation holds

\[
\frac{\partial \rho(v_{ff}; x, t)}{\partial t} + \frac{\partial}{\partial x} \left( v(v_{ff}; \rho(x, t))\rho(v_{ff}; x, t) \right) = f(v_{ff}; x, t)
\] (1)
The total density of subjects is given by the sum of densities for all free flow velocities.

\[ \rho(x, t) = \int_{0}^{v_{\text{ff}} \rho_{\text{max}}} \rho(v; x, t) dv \] (2)

The fundamental relation between velocity and density for pedestrians can be adopted according to:

\[ v(v_{\text{ff}}; \rho) = v_{\text{ff}} \left[ 1 - \exp \left( -\gamma \left( \frac{1}{\rho} - \frac{1}{\rho_{\text{max}}} \right) \right) \right] \] (3)

and for cars:

\[ v(v_{\text{ff}}; \rho) = v_{\text{ff}} \frac{\rho_{\text{max}}^n - \rho^n}{\rho_{\text{max}}^n + K \rho^n} \] (4)

The maximum density \( \rho_{\text{max}} \) and the factor \( \gamma \) are unique parameters for pedestrians, as well as \( n \) and \( K \) for cars. The differential equations for pedestrians and cars were discretized by an upwind finite volume scheme (HARTMANN and VON SIVERS, 2013).

2.3 Discretization

The used parameters are \( \rho_{\text{max}} = 5.4 \text{people/m}^2 \), \( \gamma = 1.913 \), \( v_{\text{ff}} = 1.34 \text{m/s} \) in case of pedestrians, and \( \rho_{\text{max}} = 0.12 \text{cars/m} \), \( K = 6.83 \), \( n = 1.81 \), \( v_{\text{ff}} = 15 \text{ m/s} \) for cars in the city. The following finite difference is used:

\[ \rho_i^{j+1} = \rho_i^j - \Delta t \left[ d^+ F_\Delta^+ + d^- F_\Delta^- \right] \] (5)

In this case the proper method is used:

\[ F_\Delta^+ = \frac{\rho_i v(v_{\text{ff}}; (1 - \alpha)\rho_i + \alpha \rho_{i+1}) - \rho_{i-1} v(v_{\text{ff}}; (1 - \alpha)\rho_{i-1} + \alpha \rho_i)}{\Delta x} \] (6)

\[ F_\Delta^- = \frac{\rho_{i+1} v(v_{\text{ff}}; (1 - \alpha)\rho_{i+1} + \alpha \rho_i) - \rho_i v(v_{\text{ff}}; (1 - \alpha)\rho_i + \alpha \rho_{i-1})}{\Delta x} \] (7)

Being \( \alpha \) a discretization parameter between 0 and 1, \( d^+ = 1 \) and \( d^- = 0 \) always for forward moving cars and pedestrians. When the velocity is constant, this scheme reduces to first order upwind method.

2.3.1 Simulation of structured model

At first, the initial densities in the starting nodes separate into different velocity classes. If a very big event in which the number of pedestrians is very high (central limit theorem) is assumed, a normal distribution to predict how many people has each free flow velocity class can be used, using \( N(1,34, 0,26) \) as specified in (HARTMANN and VON SIVERS, 2013)

\[ \rho_{j,0} = \rho_0 \left[ N.CumulativeProbability(v_j) - N.CumulativeProbability(v_{j-1}) \right] \] (8)
In this case, some 'lines' of people that walk are assumed, each of them with a different free flow velocity. A person never changes his or her own free flow velocity. Assuming that, the velocity can be further decomposed as:

\[ v(v_{ff}; \rho(x, t)) = v_{ff} \hat{v}(\rho(x, t)) \]  

The \( \hat{v} \) part is dependant on the total density, and the \( v_{ff} \) part is the well known normal-distributed free flow velocity for each class.

### 2.4 Mass conservation

The upwind schemes are in principle mass conservative within an infinite domain. That means as well as the values do not 'escape' through the boundaries the mass is conserved. However, the edges in this case are indeed finite so a scheme that takes people close to the boundaries before they leave and move them into other edges is necessary.

The spatial discretization of an edge has \( N \) cells, so the array goes from 0 to \( N - 1 \). The cells 0 and \( N - 1 \) are boundary conditions and do not belong to the solver domain, and their value is zero. In order to assure the mass conservation, the cell 1 (second cell) is considered the start cell so the people from the beginning node is placed there. The cell \( N - 2 \) (penultimate cell) is considered the end cell of an edge and those people are sent to the end node when they reach the cell. The penultimate cell \( N - 2 \) is then emptied there after.

When some people reach the penultimate cell of an incoming edge, those people are sent to the node. Then, when the node is computed, the remaining people are placed in the outcoming edges according to the weights. The way to split the people, and assuming not-infinite nodes that let the people accumulate in the incoming edges are aspects that should be taken into account. Integrating the densities in the edges, and then summing the people in the nodes can be useful if the user wants to check if people are preserved through the computation.

\[ \text{people} = PPC \sum_s w_s \Delta x \sum_{i=0}^{N_s-1} \rho_{s,i} + \sum_w w_w \Delta x \sum_{i=0}^{N_w-1} \rho_{w,i} + \sum_n \rho_n \]  

Where \( PPC \) means persons per car, \( s \) streets, \( w \) walkways and \( n \) nodes. In the nodes we calculate with the number of pedestrians and in the edges with the local density which is given in persons per square metre. Initializing people and cars in the start nodes is the first step in the computation.

### 2.5 Fixed distributor

In this distributor, the same density is placed for all the outcoming edges so the number of people that an edge takes is proportional to the width. This density can be found using the following equation.

\[ \sum \rho_{in} w_{in} dx = \sum \rho_{out} w_{out} dx \]
\[
\rho_{\text{out}} = \frac{\sum \rho_{\text{in}} w_{\text{in}} dx}{\sum w_{\text{out}} dx}
\]  

(12)

In the start nodes the entering people and cars are initialized using the information from the GUI. Sometimes the \( \rho_{\text{out}} \) is greater than the maximum density allowed in the edges. This is a situation that happens quite often i.e. in a sudden change of width in two consecutive edges. If the first edge is very wide, a lot of people will want to go into the following edge (in this case a very narrow one). The first solution is assuming infinite nodes, so in that case the people will wait until the second cell of the outcoming edge allows them.

### 2.6 Finite nodes

In situations like the one explained before, infinite nodes shall not be used because such behavior does not happen in reality. The people should go inside the node and then leave immediately, and if they find the end node full, they should stay in the incoming edge instead of getting inside that full node. Unfortunately, the first order equation and the structured model leads the fast people go first and always the density is a decreasing function through the edge. The model equation by itself is far away from concentrating the people close to the edge exit when the node is full. Also, the parameter \( \alpha = 1 \) (remember Hartmann and Sivers numerical method) is more ‘human-wise’ because the people can ‘look forward’ and use that density to calculate their velocity. This model and numerical method does not look beyond one cell density in the edge. Implementing a numerical method or another model with longer sight for the velocity, could make the people move in a more human-wise way.

In order to allow the people to accumulate before a full node (see Figure 2) in the incoming edge, some new routines must be implemented. Section 7 explains these algorithms in detail.

![Figure 2: An Example Showing a Full Cell](image)

Figure 2 shows the behavior of pedestrians in a narrowing example, in both infinite and finite nodes:

### 2.7 Routing Behavior of Pedestrian and Cars

In their original paper, Hartmann and Sivers used fixed probability values to distribute the pedestrians at each intersection. These values were obtained by comparing the capacities of the adjacent
edges and did not take the current density of those edges into account. It is possible that the major-
ity of a flow gets directed into an edge which is already relatively full, and the required time for the
flow to pass the system could increase in an unrealistic way. Therefore, we extend this approach
and introduce a routing algorithm which determines the possible edges the pedestrians or cars will
be allocated at.

To find the fastest path through the system, we use the classical Dijkstra's algorithm (Dijkstra,
1959). This algorithm weighs all edges of the graph to compute the shortest way through the sys-
tem. We used the Dijkstra's algorithm to calculate the route with the shortest travel time. Therefore,
we used the length and the current velocity $v(v_{ff}; \rho)$ as weighting factors:

$$Weight = \frac{\text{length}}{v(v_{ff}; \rho)}$$  \hspace{1cm} (13)

With increasing density the velocity $v(v_{ff}; \rho)$ decreases according to Equation 3. If the density of
an edge reaches the maximum density $\rho_{\text{max}}$, the edge is considered as closed. Closed edges are
ignored for the calculation of the routing algorithm.
2.8 Transformation Between The Two Simulation Models

Cars get transformed into pedestrians in the parking lot nodes. Typically, for ordinary cars, one to five people get out of each vehicle, with a prescribed random distribution. The distribution for the transformation was gathered from field data (BIEDERMANN, KIELAR, AUMANN, LAI and OSORIO, 2015). The transformed pedestrians and cars receive their individual parameters, like their free flow velocity or their maximum density, according to WEIDMANN, 1993 and WACHS, 2000 respectively.

3 Visitor Distribution of Incoming Cars

To model a realistic transformation of pedestrians and cars, it is necessary to know the distribution of passengers per car. This means, that we have to know how many visitors are normally transported by one car. Therefore, we conducted an extensive field study to obtain the lacking data. We studied a large music festival in Munich over three consecutive days and counted the amount of visitors per car. We surveyed a total number of 1960 cars (for details see Table 1), which carried an average of 2.21 persons. Over 70% of the vehicles were occupied by one or two passengers. A negligible amount of cars transported more than five people, and the maximum observed was eight passengers per car. The observed distributions from all three days can be seen in Figure 4. This data was used as a configuration for the transformation process of the hybrid model.

4 The CarPed-Toolbox

4.1 Structure and Usability

The software toolbox CarPed was developed as a proof of concept. Great attention was payed towards usability. Therefore, we implemented an intuitive and easy to use graphical user interface (GUI).
Table 1: Number of Visitors Per Car on The Music Festival Rockavaria

| Date     | 1  | 2  | 3  | 4  | 5  | 5+ | Total |
|----------|----|----|----|----|----|----|-------|
| 29.05.2015 | 144| 190| 53 | 33 | 11 | 4  | 435   |
| 30.05.2015 | 98 | 263| 62 | 54 | 15 | 3  | 495   |
| 31.05.2015 | 210| 526| 158| 98 | 36 | 2  | 1030  |
| Total     | 452| 979| 273| 185| 62 | 9  | 1960  |

The GUI is divided into three major parts, namely the network input, the model and solving options and the result window. The network, consisting of nodes and edges, can be easily entered through the network input. All network depending properties, like the length or the capacity of an edge, can be determined immediately after the input procedure. By using the model and solving options window, the user can determine which computational method should be used and how the data should be processed. Finally, the computed results can be accessed and visualized in the viewer. As all computed data is stored, the density distribution for each time-step can be looked at by the user. Different coloring of the edges shows the current densities. Alternatively, the user can click on an edge or node to get more detailed information about this object like the amount of pedestrians or cars on it. An exemplary screenshot of the GUI can be seen in Figure 3.
4.2 Workflow

The network can be entered by clicking on the designated area in the GUI. Special nodes, like entry and exit nodes or parking lots, can be defined by the user. Additionally, the nodes can be linked to incoming or outgoing edges. If the input to the system is finished, a checking routine searches for errors and discrepancies. For each algorithm, the user can individually modify the settings. After everything is set up, the solver can begin its calculations by just pressing a button.

At the beginning of the simulation, the network model reads the information of nodes and edges from the database previously created. It constructs a network from the infrastructure and passes the entire network to the numerical solver. Together with the received input data, the numerical solver starts generating vehicles and pedestrians in the associated entry nodes. It determines the initial density for each edge that is connected to the entry node and computes the density distribution. For each time-step, the results are passed from the numerical solver and stored in the network model. If a subject enters a node, it gets removed from this incoming edge and is put into an outgoing edge according to the route calculated by the Dijkstra algorithm. Additionally, the subject gets transformed if the visited node is a parking lot. The computations are repeated until all pedestrians and cars have reached an exit node and are dismissed from the system.

During the result analysis, the GUI displays two visions: a parameter set and an animation of density distributions. All statistical information is provided by the toolbox and can be easily accessed.

5 Conclusion and Outlook

This report describes the dedicated hybrid and macroscopic simulation approach CarPed. This approach uses the macroscopic Hartmann and Sivers model on a network of nodes and edges. It is able to simulate human and vehicular flow in one hybrid model. The transformation between pedestrians and cars is carried out at special nodes, which represent parking lots.

The tool is thought to play a fundamental role in the planning phase of massive public events. Organizers can use it to identify pathways and roads, which have a higher risk of congestions. Therefore, the organizers can develop strategies to avoid dangerous situations. In future studies, the current approach will be extended to a hybrid model for all-encompassing traffic control. This extended hybrid model will include more traffic subjects, like trains or ships to receive a global and universal traffic simulator.

Additionally, the CarPed simulator needs to be validated by field data. Therefore, we observed a public music festival over two consecutive years (BIEDERMANN, DIETRICH, HANDEL, KIELAR, and SEITZ, 2015). The first results are promising, but need further investigations.
6 Acknowledgements

We would like to thank our supervisors Daniel H. Biedermann and Peter K. Kielar for their nice suggestions, advices, and their unconditional help at any time. Also Isabella von Sivers for her support and long conversations according to this topic. Additionally, we would like to thank Andrea Mayer, Andreas Riedl and other student assistants for their help to collect sufficient data.

7 Program Code

7.1 Subalgorithm for Traverse(edge)

The traverse algorithm in the edge first check if the end node is full, to see if there are still people in the penultimate cell. If that’s the case, only solve the edge. If not throw the people in the penultimate cell into the node and solve the edge. The penultimate available cell should be empty before the solver, otherwise mass is not conserved. That’s why the solver (see figure 6) has inside a Distribute Last Densities function that finds and empties the available penultimate cell. After solving, check again for full node to throw people. Finally, apply traverse to the end node.

7.2 Subalgorithm Solve(Edge)

Check if there are full cells in edge. If that’s the case, distribute last densities before and after computing the timestep.
7.3 Sub algorithm for Traverse(Node)

This traverse checks if Fixed or Dijkstra distributor is chosen.

7.4 Sub algorithm for Distributor Fixed

After all the incoming edges of the node threw people into it, we have a total amount to distribute into the outcoming ones. If that density to distribute cannot fit into the outcoming edges, the node is set full and it throws only the possible amount of people into the outgoing edge, given by a friction which is nothing else that a quantity between 0 and 1, the percentage of people that fits. If density to distribute fits all, set the node not full and throw all the people into the outcoming edges.

7.5 Subalgorithm Node.SetFull(bool)

If the input is true, check first if incoming edges have full cells. If that is the case do nothing, because there the Distribute Last Densities function in the edge already took the job of reorganizing the edge cells. If they have zero full cells, set one full cell for all of them. If false, set all full cells zero.

7.6 Subalgorithm Edge.distributeLastDensities()

The routine checks at first if both ultimate and penultimate denisties can be summed, using the amount. If this amount is more than the maximum density, define a friction and move only the possible people. After that, increase by one the number of full cells, and then call the same function again recursively. If thats not the case, just move people from the penultimate cell to the ultimate one.
Figure 8: Algorithm for Traverse(Node)

Figure 9: Algorithm for Node.setFull(bool)

Figure 10: Algorithm for Edge.distributeLastDensities()
References

BIEDERMANN, D. H., DIETRICH, F., HANDEL, O., KIELAR, P. M. and SEITZ, M. (2015), Using Raspberry Pi for scientific video observation of pedestrians during a music festival., Technical report, Technische Universität München.

BIEDERMANN, D. H., KIELAR, P. M., AUMANN, Q., LAI, C. T. W. and OSORIO, C. M. (2015), Carped – a hybrid and macroscopic traffic and pedestrian simulator, in ‘Proceedings of the 27th Forum Bauinformatik’.

BIEDERMANN, D. H., KIELAR, P. M. and HANDEL, O. (2014), ‘Betrachtung der Modellierung und Simulation von Fussgaengern im Kontext verschiedener Skalen’, Proc. of the 26th Forum Bauinformatik A, 171–179.

BIEDERMANN, D. H., KIELAR, P. M., HANDEL, O. and BORRMANN, A. (2014), ‘Towards TransiTUM: A generic framework for multiscale coupling of pedestrian simulation models based on transition zones’, Transportation Research Procedia 2 B, 495–500.

BORSCHE, R., KLAR, A., KÜHN, S. and MEURER, A. (2014), ‘Coupling traffic flow networks to pedestrian motion’, Mathematical Models and Methods in Applied Sciences 24(02), 359–380.

COLOMBO, R. and ROSINI, M. (2005), ‘Pedestrian flows and non-classical shocks’, Mathematical Methods in the Applied Sciences 28(13), 1–17.

DIJKSTRA, E. W. (1959), ‘A note on two problems in connexion with graphs’, Numerische mathematik 1(1), 269–271.

GALEA, E. R., WANG, Z., VEERASWAMY, A., JIA, F., J., L. P. and EWER, J. (2008), ‘Coupled fire/evacuation analysis of the station nightclub fire’, Fire Safety Science 9, 465–476.

GÖTTLICH, S., KÜHN, S., OHST, J. P., S., R. and M., T. (2011), ‘Evacuation dynamics influenced by spreading hazardous material’, Networks and Heterogeneous Media 1, 443–464.

HARTMANN, D. and VON SIVERS, I. (2013), ‘Structured first order conservation models for pedestrian dynamics’, Networks and Heterogeneous Media 8 (4), 9851007.

HITZLER, R. (2010), Eventisierung: drei Fallstudien zum marketingstrategischen Massenspaß, in ‘Springer Verlag’.

HITZLER, R., KIRCHNER, B. and BETZ, G. (2011), ‘Das Beispiel Loveparade’, Urbane Events.Springer. 1, 261277.

HUGHES, R. L. (2002), ‘A continuum theory for the flow of pedestrians’, Transportational Research Part B: Methodological 36, 507–535.

HUGHES, R. L. (2003), ‘The flow of human crowds’, Annual review of fluid mechanics 35 (1), 169182.
IJAZ, K., SOHAIL, S. and HASHISH, S. (2015), A survey of latest approaches for crowd simulation and modeling using hybrid techniques, in ‘17th UKSIM-AMSS International Conference on Modelling and Simulation’, pp. 111–116.

LEWIS, J. (1989), A value-added analysis of the Heysel stadium soccer riot, in ‘Current Psychology’.

LIGHTHILL, M. J. and WHITHAM, G. B. (1955), ‘On kinematic waves. ii. A theory of traffic flow on long crowded roads’, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 229 (1178), 317345.

MCCREA, J. and MOUTARI, S. (2010), ‘A hybrid macroscopic-based model for traffic flow in road networks’, European Journal of Operational Research Volume 207, Issue 2, 676–684.

NGUYEN, T. N. A., ZUCKER, J. D., D., N. H. and ALEXIS D., V. D. A. (2012), Hybrid equation-based and agent-based modeling of crowd evacuation on road network, in ‘Proc. of the 8th ICCS conference’, pp. 558–570.

PRETTO, C. O., CYBIS, H. B. B. and JACOBSEN, A. (2011), A multi-layer simulation model for vehicle and pedestrian interaction, in ‘Transportation Research Board 90th Annual Meeting’.

RICHARDS, P. (1956), ‘Shock waves on the highway’, Operations research 4 (1), 4251.

TOLBA, C., LEFEBVRE, D., THOMAS, P. and EL MOUDNI, A. (2005), ‘Continuous and timed Petri nets for the macroscopic and microscopic traffic flow modelling’, Simulation Modeling Practice and Theory 13, 407–436.

WACHS, M. E. A. (2000), Highway capacity manual, in ‘Transportation Research Board’.

WEIDMANN, U. (1993), Transporttechnik der Fussgänger, in ‘Schriftenreihe des IVT Nr. 90’.