Multi agent based large-scale evacuation simulation

2008-Jul-31

Gregor Lämmel
Transport Systems Planning and Transport Telematics,
Berlin Institute of Technology, D-10587 Berlin
phone: +49-30-314 21376
fax: +49-30-314 26269
laemmel@vsp.tu-berlin.de

Kai Nagel
Transport Systems Planning and Transport Telematics,
Berlin Institute of Technology, D-10587 Berlin
phone: +49-30-314 23308
fax: +49-30-314 26269
nagel@vsp.tu-berlin.de

Words: 6656 + 3 figures

ABSTRACT

The evacuation of whole cities or even regions is an important problem, as demonstrated by recent events such as the evacuation of Houston in the case of Hurricane Rita or the evacuation of coastal cities in the case of tsunamis. A robust and flexible simulation framework for such large-scale disasters helps to predict the evacuation process. Furthermore, it is possible to recognize bottlenecks in advance, so that an elimination of those bottlenecks is possible. This should lead to a better preparedness for an event of evacuation for cities or regions that face a high risk of natural disasters. Existing methods are either geared towards smaller problems (e.g. Cellular Automata techniques or methods based on differential equations) or are not microscopic (e.g. methods based on dynamic traffic assignment). This paper presents a technique that is both microscopic and capable to process large problems. The simulation was applied to the Indonesian city of Padang. The city faces a high risk of being inundated by an earth quake triggered tsunami.

INTRODUCTION

The evacuation of whole cities or even regions is an important problem, as demonstrated by recent events such as the evacuation of Houston in the case of Hurricane Rita or the evacuation of coastal cities in the case of tsunamis. One example for a highly vulnerable area is the city of Padang, West Sumatra, Indonesia. Sumatra's third largest city is located directly on the coast and partially sited beneath the sea level, and thus is located in a zone of extreme risk due to severe earthquakes and tentatively triggered tsunamis. The city of Padang has been hit by tsunamis in the past. The most well documented tsunamis are the ones from 1797 and 1833 *(1)*. Both tsunamis inundated large parts of the city. However, these past tsunami events are hardly comparable with the current local situation due to major changes in land use pattern. What is more, population figures have risen strongly. Today the city has approx. 1,000,000 inhabitants and the most densely populated parts of the city are located directly at the shore line.

The "Last-Mile Evacuation" research project *(2, 3)* develops a numerical last mile tsunami early warning and evacuation information system on the basis of detailed earth observation data and techniques as well as unsteady, hydro-numerical modeling of small-scale flooding and inundation dynamics of the tsunami, combined with evacuation simulations in the urban coastal hinterland for the city of Padang.

Since the advance warning time before the tsunami wave reaches the coast line is only 20-40 minutes, the evacuation must be as quick as possible. Even if not all of the estimated 1,000,000 inhabitants need to be evacuated, the number of evacuees could be hundreds of thousands. Therefore a detailed analysis of aspects that could influence the evacuation process is necessary. With this analysis it should be possible to:

- Give an estimate of the evacuation time.
- Detect bottlenecks that could for example emerge at bridges.
- Detect highly endangered areas, where a vertical evacuation[1] seems the only way.

Because of the complexity of the system, an analytic solution to this problem seems to be hard. Therefore a microscopic multi-agent simulation for the city with all its inhabitants has been developed.

RELATED WORK

Congruent with the importance of the topic, there is a large body of research regarding emergency evacuations. As a first classification, one may differentiate between two situations: (i) evacuation from within buildings, ships, airplanes, etc.; (ii) large-scale citywide or regional evacuations, e.g. because of nuclear power plants failures or because of hurricanes. Case (i) case usually concerns pedestrian evacuation; case (ii) usually uses traffic-based evacuation.

A good overview of pedestrian evacuation modeling and software can be found in the books of the bi-annual conference series "Pedestrian and Evacuation Dynamics" *(4, 5, 6, 7)*. Pedestrian evacuation simulations can be classified into microscopic and macroscopic ones. Microscopic models represent space, time, and persons on a fine-grained level. Possible microscopic approaches are Cellular Automata (CA) *(8)*, discretized differential equations ("molecular dynamics (MD)") *(9, 10)*, or movement rules based on random utility modelling *(11)*. Examples of software packages based on microscopic models are Exodus *(12)*, Myriad (www.crowddynamics.com), Egress

[1] quake and tsunami proof shelters

(www.aeat-safety-and-risk.com/html/egress.html), and PedGo *(13)*. Macroscopic models use the analogy of flows of pedestrians and liquids. Examples of software packages based on macroscopic models are Aseri *(14)* and Simulex (www.iesve.com). See Refs. *(15)* and *(16)* for surveys. Compared to what is known in terms of field measurements (e.g. *(17, 18)*), most if not all packages lead to similar results *(19)*.

Once the pedestrian movement model is selected, it is necessary to define the evacuation directions. For more complex geometries, this is no longer a single movement towards one or two exits, but may involve rather complex movments in a building or in a street network. The arguably simplest solution is a grid-based potential function where the "uphill direction" leads to the nearest exit *(20)*. The same can be done using essentially continous spatial variables, at the expense of much larger computing times *(21)*. Alternatively, routing can be done along graphs *(22, 23)*, which is a much faster technique when the abstraction to a graph is possible.

Another line of research concerns citywide or regional evacuations, i.e. case (ii). The development of these tools was much influenced by the development of tools in the areas of transport planning and traffic management. At the core of many of these methods is a static assignment routine (e.g. *(24, 25)*). A typical example for traffic-based evacuation simulation based on static concepts is MASSVAC *(26)* although later versions contain dynamic aspects.

A severe shortcoming of static assignment is that it does not possess any consideration of the time-of-day dynamics. Dynamic traffic assignment (DTA) is defined as a distribution of time-dependent trips on routes. A typical approach to implement DTA is day-to-day re-planning: The traffic flow simulation (also called network loading) is run with pre-specified routes, route costs are extracted, some or all of the routes are modified, the traffic flow simulation is run again, etc., until some stopping criterion is fulfilled. Examples of stopping criteria are that either every trip uses a route which minimizes expected travel time (time-dependent Nash equilibrium), or it selects between different route alternatives following a pre-specified distribution function (time-dependent SUE).

Many DTA packages have been tested in the evacuation context: MITSIM *(27)*, Dynasmart *(28, 29)*, PARAMICS *(30)*, and VISSIM *(31)*. Oak Ridge National Laboratory has a package named "OREMS" (cta.ornl.gov/cata/One_Pagers/OREMS.pdf) explicitly for evacuation traffic. Publications stressing dynamic aspects of traffic-based evacuation as a novelty can be found as recent as 2000, e.g. *(32, 33)*. For a review, see *(34)*.

A further distinction is if travelers can re-route while they are on their way (within-day re-planning; en-route re-planning), or only before their trip (day-to-day re-planning; pre-trip re-planning) *(35)*. Clearly, en-route re-planning capability is more realistic. It is, however, also more demanding: Adaptation of the plans needs to be called frequently from within the network loading, rather than only having to alternate between the network loading and the mental layers as one does in day-to-day re-planning.

A large body of work (e.g. *(36, 37)*) uses microsimulation to investigate the issues of contraflow evacuation, i.e. the reversal of inbound lanes of a freeway in order to obtain additional outbound capacity.

To our knowledge, none of the above aproaches is able to simulate large-scale scenarios (with millions of entities) while remaining microscopic:

- With a CA model, an area of $40\,km \times 40\,km$ translates into 10^{10} cells. Even if every cell only needs 1 Byte, this still translates into 10 GByte of memory, resulting in large simulation times.

- For the MD approach, the problem are the sub-second time resolutions that are typically used *(38)*.
- DTA approaches seem the most likely candidates, but to our knowledge their implementation of the traffic flow dynamics usually is still too time-consuming for scenarios of that size.

One way to achieve faster computation with a microscopic model is to use a model with deliberately large time steps and to computationally concentrate on those areas (links) where the pedestrian movement actually takes place *(39)*. Another approach is based up on a modified queuing model *(40, 41)*. The queuing model simplifies streets to edges and crossings to nodes; the difference to standard queuing theory is that agents (particles) are not dropped but spill back, causing congestion. This graph-oriented model is defined by lengths/widths, free speed and flow capacity of the edges. This simplification leads to a major speedup of the simulation while keeping results realistic The combination of these two approaches (switching off unused links; queue model) is used in this paper.

SIMULATION FRAMEWORK

The simulation framework is based on the MATSim framework for transport simulation *(42)*. Since MATSim is focused on simulation of motorized traffic, several modifications were necessary. The key elements are:

- The agent database, where every agent represents one evacuee.
- The simulation network, based on links and nodes.
- The traffic flow simulator, where all the agents plans are executed.
- The plans generator, which generates an escape plan for every agent.
- There is a mechanism that allows improving the performance of the agents' plans by repeatedly trying to find faster evacuation routes.

In the following section we will give an overview how MATSim works in general but the focus will be on the pedestrian specific modification or extension of MATSim.

Synthetic population, plans, agent database

MATSim always starts with a synthetic population. A synthetic population is a randomly generated population of individuals which is based as much as possible on existing data such as census data. For evacuation, the synthetic population is the collection of all synthetic individuals that are involved in the evacuation.

Every synthetic individual possesses one or several plans. These plans are "intentions" of the synthetic individuals, to be tested in the traffic flow simulation (described later), and scored afterwards. For evacuation, the plans are evacuation strategies. For example, such a strategy may be to leave the building 5 minutes after a second warning, and follow a predetermined route to safety. The collection of agents together with their plans is sometimes called an agent database.

People can have different positions within the city when a warning occurs. For example, they can be at home or at work. Therefore, also in the evacuation context it makes sense to consider MATSim plans in their more conventional meaning, as a description on what a synthetic traveller intends to do during a normal day. One can then run a regular traffic flow simulation with these plans, stop it at the time of an evacuation warning, and use the positions of all agents at the time of that warning as the initial condition to the evacuation. This will be the subject of future work.

Simulation network

The simulation network represents the area that is accessible by the evacuees. In the case of a vehicular evacuation this network consists of all accessible streets. Each street segment defines a link. The parameters of the links are the length, capacity and the free flow speed. For a pedestrian evacuation the links in the simulation network also consist of squares and sidewalks. The flow capacity is given by the width of a link as described in the next section. A good way of creating the simulation network is by extracting the needed information from satellite imagery. In this project we rely on IKONOS imageries. The extraction has been done by using an object-oriented hierarchical classification approach *(43)*. After the raw street map was extracted, it was decomposed it into crossings and street segments. An exemplary picture of the generated street segments is shown in Fig. 1 a). To make the segments usable for the physical evacuation simulation, they have to be converted into a logical graph. This has been done by converting streets to links and crossings to nodes, and connecting adjacent links to nodes and vice versa. A graph representation of the street network is shown in Fig. 1 b).

In former work *(3)* we reported some difficulties with medians. The extraction algorithm classifies streets based on the surface conditions and will consequently produce two parallel unconnected links if there is a median. If in that network an evacuee wanted to flee from one side of the street to the other side she had to make a detour. There is no simple solution for this problem, since it is not possible to detect obstacles (e.g. fences) on the median, and so it is not clear if the median is traversable by pedestrians. In the meantime we obtained photographs of some of the medians and we could detect traversing options by manually looking over satellite imagery elsewhere. The improved network leads to much better evacuation routes. This issue is discussed later in the *Results* section.

Traffic flow simulator

The traffic flow simulation is implemented as a queue simulation, where each street (link) is represented as a FIFO (first-in first-out) queue with three restrictions *(40)*. First, each agent has to remain for a certain time on the link, corresponding to the free speed travel time. Second, a link flow capacity is defined which limits the outflow from the link. If, in any given time step, that capacity is used up, no more agents can leave the link. Finally, a link storage capacity is defined which limits the number of agents on the link. If it is filled up, no more agents can enter this link. The difference to standard queueing theory is that agents (particles) are not dropped but spill back, causing congestion. An illustration of the queue model is shown in Fig. 1 c). The parameters of the model are:

- Link minimum width w
- Link area A
- Link length l
- Flow capacity $FC = w * C_{max} = w * 1.3\frac{p}{m*s}$
- Free flow speed $v_{max} = 1.66\frac{m}{s}$
- Storage capacity $SC = A * D_{max} = A * 5.4\frac{p}{m^2}$

where C_{max} is the maximum flow capacity per unit width, and D_{max} is the maximum density per unit area.

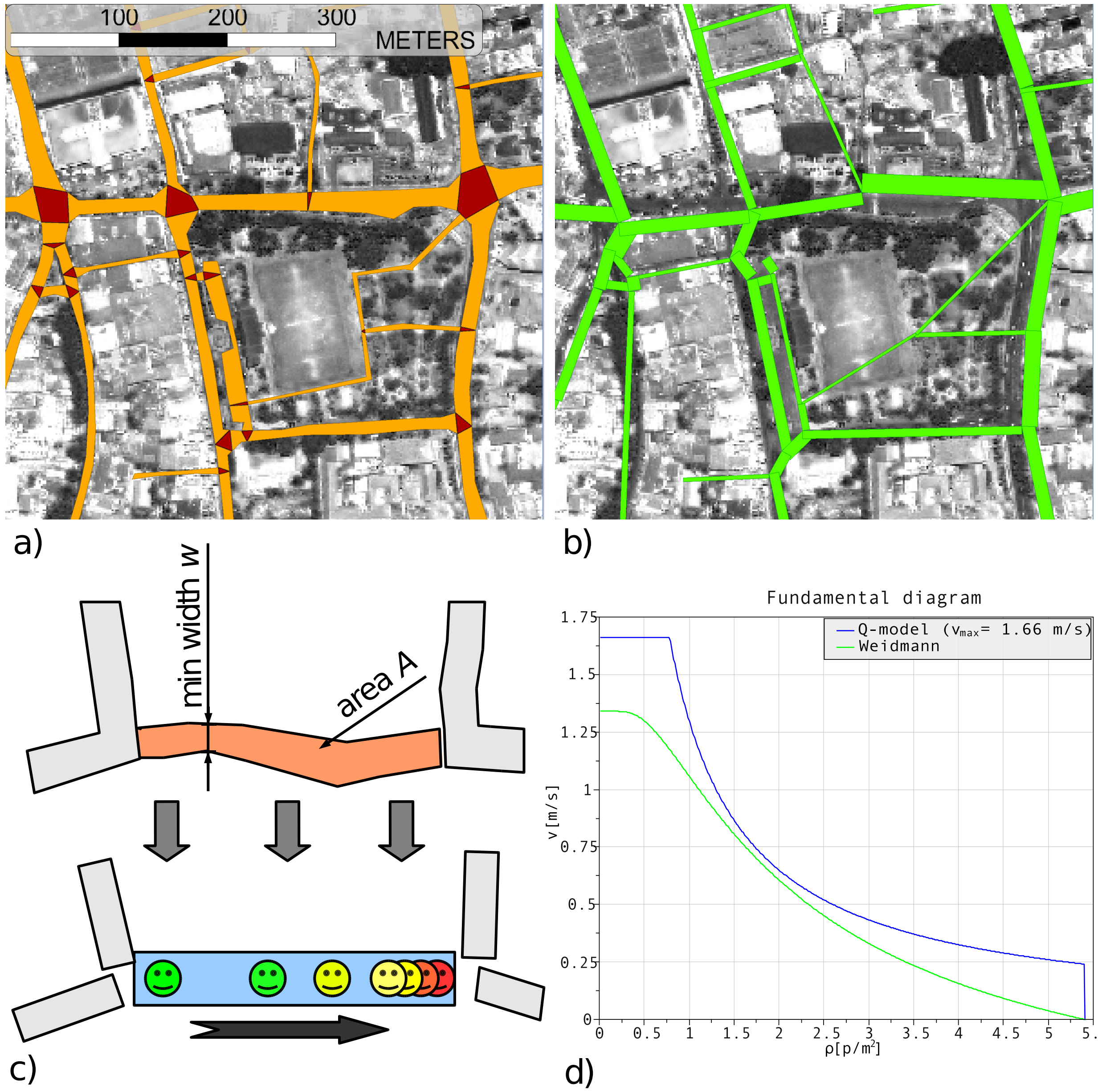

FIGURE 1 Street map extracted from Ikonos imagery (a) and the simulation network derived from this street map (b). Capacity restrictions in the queue model stem entirely from the links; in consequence, intersections are not modeled explicitly. Functioning of the queue model is shown in c) and its corresponding fundamental diagram in(d). Satellite imagery and raw street map data by the German Aerospace Center, Oberpaffenhofen (2007)

The parameters have been chosen to approximate Weidmann's fundamental diagram *(18)*.[2] He pointed out that the relation between density and velocity is adequately captured by the so-called Kladek-formula: $v_{F,hi}(D) = v_{F,hf} \times [1 - e^{-\gamma \times (\frac{1}{D} - \frac{1}{D_{max}})}]$, where $v_{F,hi}(D)$ the velocity $[m/s]$ at a

[2]Newer studies *(44)* imply other fundamental diagrams then those from Weidmann. An adaptation of these values could, in consequence, become necessary in future.

particular density D, $v_{F,hf}$ is the velocity $[m/s]$ at free flow , γ is a free parameter $[persons/m^2]$, and D_{max} the density at which no flow occurs $[persons/m^2]$. Empirical studies showed the best results with $\gamma = 1.913\, persons/m^2$, $v_{F,hf} = 1.34\, m/s$ and $D_{max} = 5.4\, person/m^2$.

Our study uses the same maximum density as Weidmann, but the free flow speed was set to $1.66\ m/s$. This value is slightly higher then the $1.34\ m/s$ used by Weidmann, but the values presented by Weidmann reflect the pedestrian flow under normal conditions and not in a case of emergency. Our queuing model, however, generates a speed-density relationship of the form $v = \min[v_{max}, FC/D]$ *(45)*. Even if a complete agreement is thus not possible, the flow dynamics produced by our queue model is not too far away from Weidmann's fundamental diagram (cf. Fig. 1 d)). Furthermore, it is important to note that our model is designed in a way that it enforces the maximum flow of FC. There have been many empirical studies investigating the dependency between flow and width of a bottleneck *(17, 46, 47)*. The proposed value of 1.3 persons per second per crossectional meter matches these empirical results. It is generally assumed that these quantities do not change with the width of the track *(44)*. Overall, the selected capacity of $1.3\frac{p}{m*s}$ seems to be a rather solid quantity, and we expect that other aspects such as parked/left vehicles will play a larger role.

The fundamental diagram produced by the queue model (Fig. 1 d)) shows that the velocity will be reduced as a link will become more and more congested. This means that congestion/spillback effects do not need to be added, but are produced by the model itself.

Plans generation

Within the MATSim framework a shortest path router based on Dijkstra's shortest path algorithm *(48)* has been implemented. This router finds the shortest path in a weighted graph from one node to any other, whereby the actual weights for a link are defined by a time-dependent cost function. Since we want to evacuate the city as fast as possible, the weights represents the (expected) travel time.

There is, however, no particular node as the target of the shortest path calculation, as the evacuees have more than one safe place to run to. Instead, in the underlying domain every node outside the evacuation area is a possible destination for an agent that is looking for an escape route. To resolve this, the standard approach (e.g. *(49)*) is to extend the network in the following way: All links which lead out of the evacuation area are connected, using virtual links with infinite flow capacity and zero length, to a special "evacuation node" (see Fig. 2). Doing so, Dijkstra's algorithm will always find the shortest route from any node inside the evacuation area to this evacuation node, and thus to the nearest exit.

Initial plans use the shortest path according to free speed travel time. In subsequent iterations (see below), link travel times are based on the previous iteration.

Agents learning

At the end of every iteration, every agent will score the performed plan. The score of a plan is the negative of its execution time (i.e. the negative of the needed time to evacuate). The scored plans remain in the agents' memory for further executions. For the learning procedure, two different learning strategies were applied. The ReRoute strategy generates new plans with new evacuation routes based on the information of the experienced travel times from the last run. This uses the router described in the previous section, with time-variant link travel times as link costs. The other strategy is called ChangeExpBeta. This strategy decides if the just performed plan should be used again, or if a random plan out of the memory should be selected for the next iteration. The probability to change the selected plan is calculated according to

$$p_{change} = min(1, \alpha * e^{\beta*(s_{other} - s_{current})/2}) \ ,$$

where

- α: The probability to change if both plans have the same score
- β: A sensitivity parameter
- $s_{\{other,current\}}$: The score of the other/current plan

In the long run, this model produces steady state probabilities according to

$$p_j = \frac{e^{\beta*s_j}}{\sum_i e^{\beta*s_i}} \ ,$$

where p_i is the probability for plan i to be selected and s_i its current score. That is, the probability distribution converges to a logit distribution, but with ChangeExpBeta only small numbers of agents change from one plan to another which results in a smoother learning curve and lets the system better convert to a steady state.

After re-planning every agent has a selected plan that will be executed in the next iteration. Repeating this iteration cycle of learning, the agents' behavior will move towards a Nash equilibrium. If the system were deterministic, then a state where every agent uses a plan that is a best response to the last iteration would be a fixed point of the iterative dynamics, and at the same time a Nash Equilibrium since no agent would have an incentive to unilaterally deviate. Since, however, the system is stochastic, this statement does not hold, and instead we look heuristically at projections of the system. From experience it is enough to run 100 iterations until the iterative dynamics has reached a steady state.

SCENARIO

The evacuation procedure of a city depends on the distribution of the population. The distribution of the population changes over the time of day. That means one has to develop different evacuation scenarios for different times of day. Here we present a scenario called: "evacuation at 3 am". That means we assume all inhabitants are at home. It is then straightforward to derive the necessary information about the population from existing census data. The presented simulation relies upon the census for 2005 and was provided by the statistical bureau of Padang *(50)*. Another important aspect is the area that has to be evacuated. Although we have obtained first results of an inundation simulation *(3)* with detailed flooding information, for this paper we stick with a more simple approach. We defined all areas with an elevation of more then $10\ m$ as safe and all other areas as unsafe. With this approach, we most likely overestimate the area that has to be evacuated. Better definitions of the endangered area will be possible when more results of the inundation dynamics, especially concerning worst case situations, will be available *(51)*.

A view of the endangered area including the borders of the sub-districts is given in Fig. 2. The area with an elevation of less then $5\ m$ is colored in red and the area with an elevation between $5\ m$ and $10\ m$ is colored orange. For every person that lives within this area an agent is generated. In the end, the database consists of 321,281 agents. This set of agents and locations builds our start setup for the evacuation. The simulation network covering this area consists of 6,289 nodes

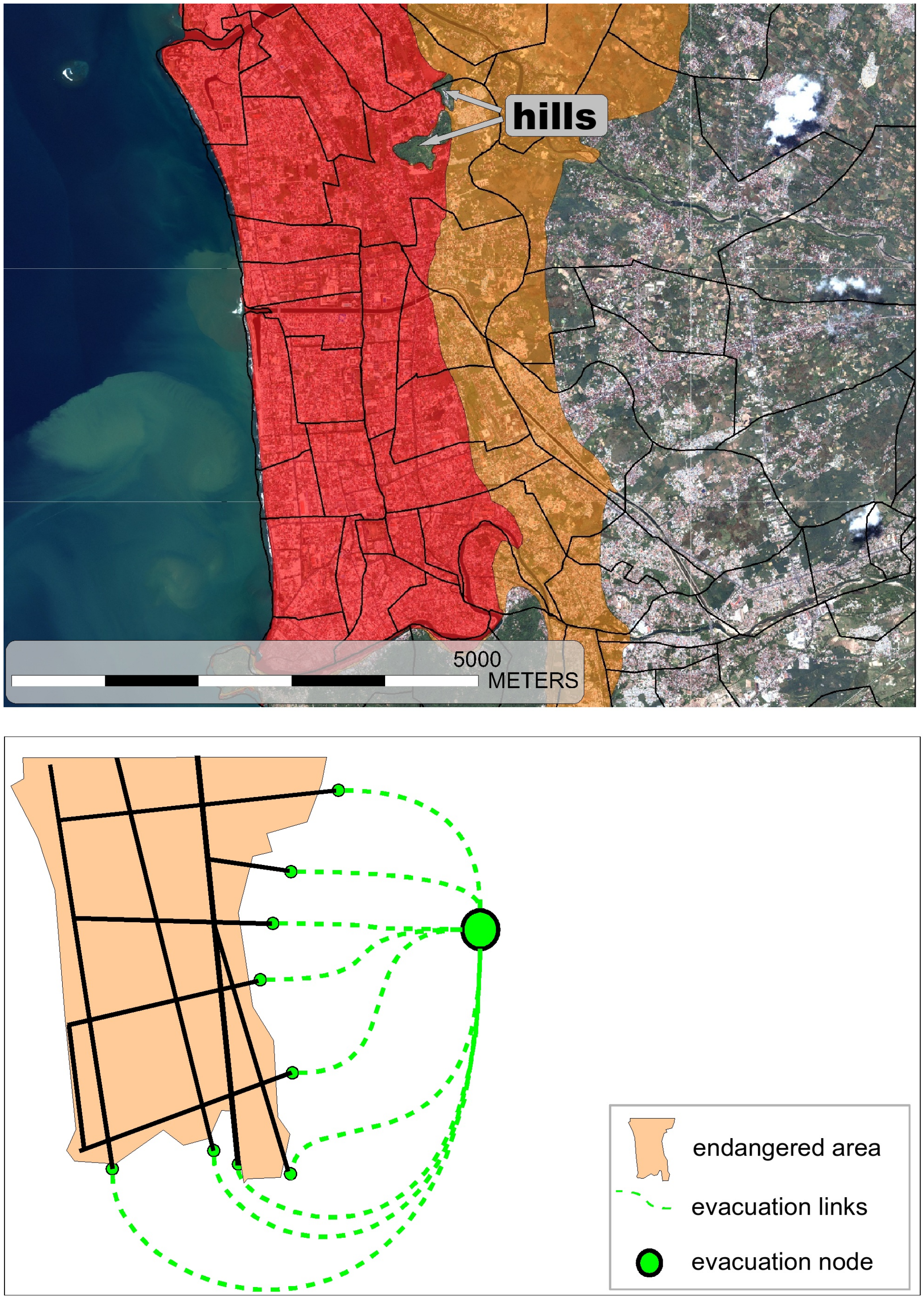

FIGURE 2 The satellite imagery (top) of the city shows district borders and (highlighted) the evacuation area. The area with an elevation of up to $5\ m$ is colored in red, and $5\ m$ to $10\ m$ in orange. Bottom: A sketch of the evacuation network with the evacuation node as super sink and virtual (i.e. zero cost) evacuation links. Satellite imagery by the German Aerospace Center, Oberpaffenhofen (2007)

and 16,978 unidirectional links. The simulation is stopped after 100 iterations of learning. As explained above, we applied two different strategies for learning to the simulation. The ReRoute strategy finds a new evacuation route for an agent, based on the experienced travel times of the former iteration. The ChangeExpBeta strategy implements a discrete-choice model that assigns a plan from the agent's memory with a probability depending on the score of the plan.

The simulation setup was as follows:

- For iteration 1 – 20: Each agent had a chance of 20% for being chosen for ReRoute and 80% for ChangeExpBeta.
- For iteration 21 – 50: Each agent had a chance of 10% for being chosen for ReRoute and 90% for ChangeExpBeta.
- For iteration 51 – 100: Only the ChangeExpBeta strategy was enabled.

This setup gives a fair arrangement between exploration and exploitation. That means in the first iterations the probability of trying out another route is much higher then in the later iterations. If the probability for ReRoute (i.e. exploration) is too low or reduced too early, then it could be that some promising routes will never be discovered. On the other hand, if the probability for ReRoute is very high during all iterations, the system tends to fluctuate and will not converge to a steady state. The system would change so fast that the agents would not get a chance to exploit their knowledge about the system.

RESULTS

The simulation run was performed on a standard computer with a CPU at 3 GHz and 3 GB of RAM. The computer runs JAVA jdk1.6_03 on Linux. The evacuation simulation was stopped after 100 iterations. An iteration took about 4 minutes including physical simulation and re-planning and the overall runtime was about 5 hours. As expected, the evacuation time decreases significantly with the iterations. Especially within the early iterations, it drops very fast. A diagram that represents this process is shown in Fig. 3 (top left).

After these 100 iterations of learning the evacuation of the endangered area took about one hour. In preliminary runs *(3)* the evacuation took about two hours. This intense decrease is caused by solving problems with the simulation network (e.g. by making medians traversable for pedestrians) The overall time alone tells little about the evacuation process itself. Therefore we evaluated the evacuation process for iteration 0, 1, 5, 10 and 100 in detail. Figure 3 (top right) shows the results.

The initial iteration results in a steep gradient (high outflow) at the beginning but it flattens very fast. As the iterations progress the initial gradient gets even steeper and becomes more linear. This indicates that there is probably a maximum cross-sectional capacity from the endangered to the safe area, and that the simulation finds a way to maximize its use. The consequence would be that it is not possible to evacuate the whole endangered area within less than about one hour. It also indicates that the impediment to faster evacuation is this cross-sectional capacity. These aspects will also be discussed below.

Some snapshots of the first $45\ min$ in the 100th iteration are shown in Fig. 3 (bottom). The agents are colorized depending on the time they need for the evacuation. A green color indicates that they escape very quickly while a red color means they need rather long time for the evacuation. One notices that the evacuation time for the individual agent mainly depends on the distance to the safe area. This indicates that no part of the city has significantly more bottlenecks, which might cause more congestion than in other parts. The only exception is the area in the North-East around

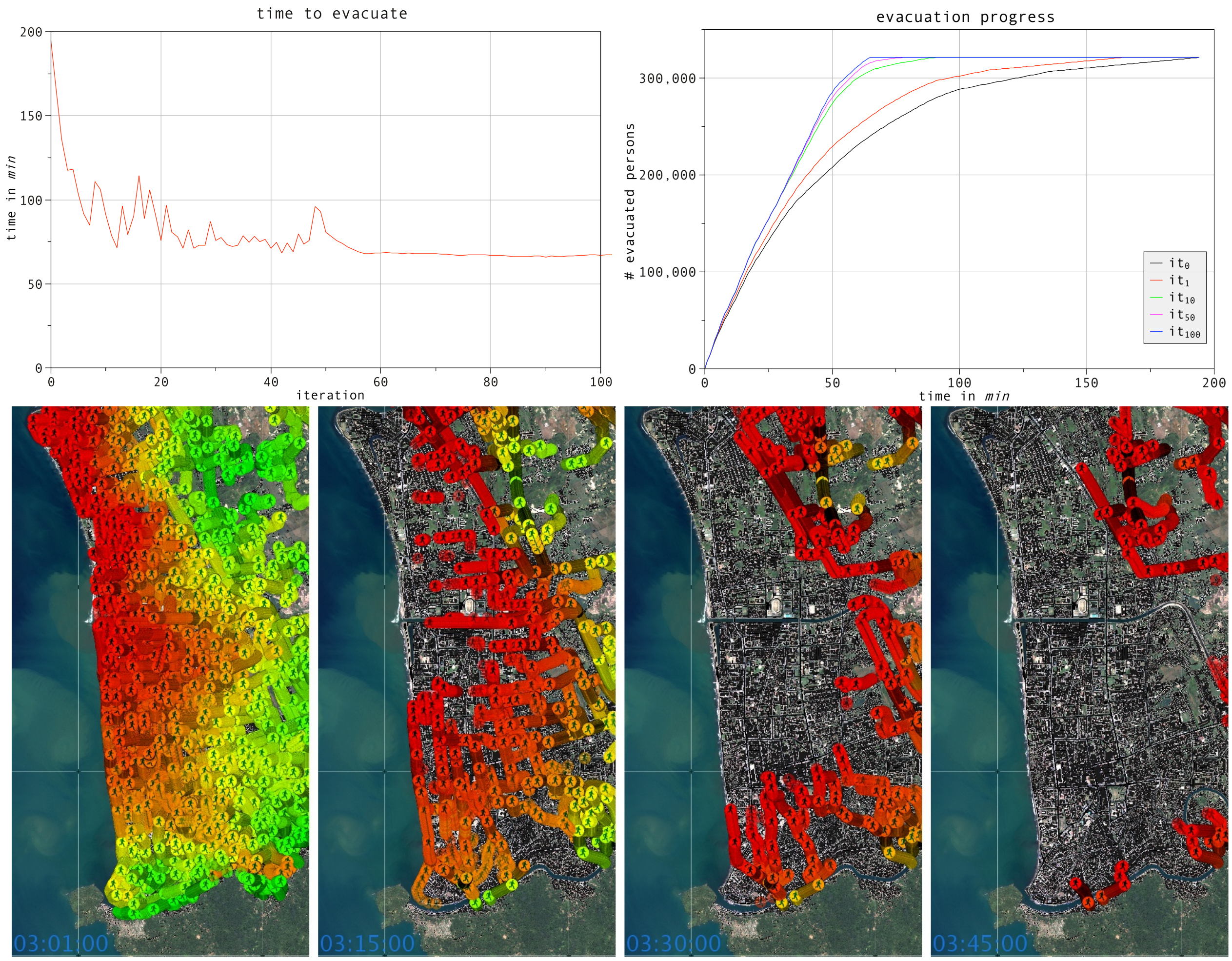

FIGURE 3 Top: Evacuation time vs. iteration number (left) and evacuation progress showing the number of evacuated people as a function of time for various iteration numbers (right). Bottom: Visualizer snapshots with time dependent colorized agents after 100 iterations of learning. Satellite imagery by the German Aerospace Center, Oberpaffenhofen (2007)

the hills (cf. Fig. 2). Many agents (approx. 100,000) evacuate to these hills causing congestion. This congestion could be reduced if the accessibility to those hills were improved.

Generally, it seems that agents are equally distributed over all evacuation points. This could be an indicator that the found solution is not to far away from an optimal solution. However, optimality is hard to verify. A lower bound is the free speed travel time of the agent with the longest distance to the safe area. In this case the agent with the longest distance to a safe area needs to walk $4742\ m$. As discussed earlier the free speed is $1.66\ m/s$ for all links. Therefore, the lower bound for an optimal solution is about 48 min. With the above discussed results we are not so far away from this value.

DISCUSSION

The iterations start from a solution where all agents take the fastest path to safety, and iterate to a stochastic version of the Nash equilibrium. The fact that the number of evacuated persons per time unit increases during the iterations (Fig. 3 (top right)) indicates that the initial solution is overly congested on some evacuation paths, and some evacuees are better off taking a longer route.

There might also be a system optimal solution, evacuating agents even faster, but forcing some agents to do something that is not optimal for themselves. Compared to the system optimal solution, the Nash equilibrium solution has the advantage that one could attempt to reach it by training: Since the solution is constructed in a way that nobody could (unilaterally) gain by deviating from this solution, there might be a chance to convince people that it is in their self-interest to follow that solution.

Yet clearly, this can only be considered as a benchmark solution. In emergency situations, people tend to be irrational and to display herd behavior *(52)*, or they might want to re-unite the family before they evacuate, causing counter-flows. Still, given a warning time of about $30\ min$, even the "rational" Nash equilibrium solution does not seem to leave enough time. However, the situation is more complicated:

- The preliminary inundation simulations indicate that our evacuation area is too large for most situations, i.e. the tsunami wave will not reach that far. A problem here, however, is that even if one assumes a functioning warning system, it will probably not include the tsunami wave height, and so a tailored evacuation seems not possible.
- At the same time, it seems impossible to implement an evacuation scheme that makes people evacuate for about an hour when this is not necessary in most cases: The compliance rate will not be very high.

Tsunami proof shelters for vertical evacuation could be a solution for those areas from where horizontal evacuation takes a long time. Since the local government in Padang plans to build some kind of shelters for vertical evacuation, one could use the simulation to find appropriate locations for these shelters. It might also be possible to use the roofs of stable buildings for shelter.

Another issue concerns the mode choice: The investigation assumes that all evacuation is done by foot while it might be reasonable to assume that some people use cars or cycles, and they might even leave vehicles in the street to continue on foot if progress by vehicle becomes too slow. For the time being, such issues are not considered. The queue model could, to a certain extent, be parameterized to deal with mixed traffic, as long as all modes move with the same speed. The effect of "stranded" vehicles could be included by a parameterization of the flow capacity of the queue model, although a behavioral model for abandoning vehicles would be needed. Beyond that, one would arguably need to switch to a true two-dimensional model such as *(52)* or *(13)*. Such models could still operate on networks *(23)*.

CONCLUSIONS

We introduced a microscopic pedestrian simulation framework for large-scale evacuations. It is implemented as a Multi Agent Simulation, where every agent tries to optimize its individual evacuation plan in an iterative way. The simulation framework is demonstrated through a case study based on a hypothetical tsunami inundation of the Indonesian city of Padang. The runtime performance shows that this approach is well suited for large scale scenarios. Despite of the underlying behavioral model being quite simple the simulation gives plausible results regarding the predicted evacuation process. The development of different evacuation scenarios for different times of day will be the topic on future work. We have already results from a household survey that took place in April/May. Currently we are working on the demand generation for the city based on this survey of 1000 households and existing census data. Later it is planned to derive the time dependent distribution of the population directly from the demand.

ACKNOWLEDGMENTS
This project was funded in part by the German Ministry for Education and Research (BMBF), under grants numbers 03G0666E ("last mile") and 03NAPI4 ("Advest"). We thank Hubert Klüpfel for fruitful discussion about pedestrian evacuation simulations, and the group of Prof. F.A. Ismail for providing the local photographs for the obstacle analysis.

REFERENCES

1. D.H. Natawidjaja, K.Sieh, M. Chlieh, J. Galetzka, B.W. Suwargadi, H. Cheng, R.L. Edwards, J-P. Avouac, and S.N. Ward. Source parameters of the great Sumatran megathrust earthquakes of 1797 and 1833 inferred from coral microatolls. *Journal of Geophysical Research*, 111, 2006.

2. J. Birkmann, S. Dech, G. Hirzinger, R. Klein, H. Klüpfel, F. Lehmann, C. Mott, K. Nagel, T. Schlurmann, N. Setiadi, F. Siegert, and G. Strunz. Numerical last-mile tsunami early warning and evacuation information system. In Ludwig Stroink, editor, *GEOTECHNOLOGIEN Science Report No. 10: "Early Warning Systems in Earth Management"*, Technical University Karlsruhe, October 2007. Die Deutsche Biliothek.

3. G. Lämmel, M. Rieser, K. Nagel, H. Taubenböck, G. Strunz, N. Goseberg, T. Schlurmann, H. Klüpfel, N. Setiadi, and J. Birkmann. Emergency preparedness in the case of a tsunami – evacuation analysis and traffic optimization for the Indonesian city of Padang. In *Pedestrian and Evacuation Dynamics (7)*.

4. M. Schreckenberg and S. D. Sharma, editors. *Pedestrian and Evacation Dynamics*. Proceedings of the 1st international conference, Duisburg, 2001. Springer, 2002.

5. E. R. Galea, editor. *Pedestrian and Evacuation Dynamics*. Proceedings of the 2nd international conference, London, 2003. CMS Press, University of Greenwich, UK, 2003.

6. P. Gattermann, N. Waldau, and M. Schreckenberg, editors. *Pedestrian and Evacuation Dynamics*. Proceedings of the 3rd international conference, Vienna. Springer, Berlin, 2006.

7. *Pedestrian and Evacuation Dynamics*. Proceedings of the 4th international conference, Wuppertal, 2008. Springer, Berlin, 2008.

8. H. Klüpfel, T. Meyer-König, A. Keßel, and M. Schreckenberg. Simulating evacuation processes and comparison to empirical results. In M. Fukui et al, editor, *Traffic and granular flow '01*, pages 449–454. Springer, Berlin Heidelberg New York, 2003.

9. D. Helbing, I.J. Farkas, P. Molnar, and T. Vicsek. Simulation of pedestrian crowds in normal and evacuation situations. In Schreckenberg and Sharma *(4)*, pages 21–58.

10. D. Helbing, L. Buzna, A. Johansson, and T. Werner. Self-organized pedestrian crowd dynamics: experiments, simulations and design solutions. *Transportation Science*, 39:1–24, 2005.

11. M. Bierlaire, G. Antonini, and M. Weber. Behavioral dynamics for pedestrians. In K.W. Axhausen, editor, *Moving through nets: The physical and social dimensions of travel*. Elsevier, 2003.

12. E.R. Galea. Simulating evacuation and circulation in planes, trains, buildings and ships using the EXODUS software. In Schreckenberg and Sharma *(4)*, pages 203–225.

13. H. Klüpfel. The simulation of crowd dynamics at very large events – Calibration, empirical data, and validation. In Gattermann et al. *(6)*.

14. V. Schneider and R. Könnecke. Simulating evacuation processes with ASERI. In Schreckenberg and Sharma *(4)*, pages 303–314.

15. M. Jafari, I. Bakhadyrov, and A. Maher. Technological advances in evacuation planning and emergency management: Current state of the art. Final Research Reports EVAC-RU4474, Center for Advanced Infrastructure and Transportation (CAIT), Rutgers University, NJ, 2003.

16. E. Kuligowski. Review of 28 egress models. Technical report, National Institute of Standards and Technology (NIST), Gaithersburg, MD, 2004.

17. W. Predtetschenski and A. Milinski. *Planning for Foot Traffic in Buildings*. Amerind Publishing Co. Pvt. Ltd., New Delhi, 1978.

18. U. Weidmann. *Transporttechnik der Fussgänger*, volume 90 of *Schriftenreihe des IVT*. Institute for Transport Planning and Systems ETH Zürich, 2 edition, 1993. In German.

19. C. Rogsch. Vergleichende Untersuchungen zur Simulation von Personenträmen. Diploma thesis, Universität Wuppertal, 2005.

20. K. Nishinari, A. Kirchner, A. Nazami, and A. Schadschneider. Extended floor field CA model for evacuation dynamics. *IEICE Transactions on Information and Systems*, E87-D(3):726–732, 2004.

21. S.P. Hoogendoorn, P.H.L Bovy, and W. Daamen. Microscopic pedestrian wayfinding and dynamic modelling. In Schreckenberg and Sharma *(4)*, pages 123–154.

22. S.A. Hamacher, H.W. Tjandra. Mathematical modelling of evacuation problems: A state of art. *Berichte des Fraunhofer ITWM*, 24:1–38, 2001.

23. C. Gloor, P. Stucki, and K. Nagel. Hybrid techniques for pedestrian simulations. In *Cellular automata, Proceedings*, number 3305 in Lecture Notes in Computer Science, pages 581–590. Springer, 2004.

24. Y. Sheffi. *Urban transportation networks: Equilibrium analysis with mathematical programming methods*. Prentice-Hall, Englewood Cliffs, NJ, USA, 1985.

25. J. de D. Ortúzar and L.G. Willumsen. *Modelling transport*. Wiley, Chichester, 1995.

26. A.G. Hobeika and C. Kim. Comparison of traffic assignments in evacuation modeling. *IEEE Transactions on Engineering Management*, 45(2):192–198, 1998.

27. M. Jha, K. Moore, and B. Pashaie. Emergency evacuation planning with microscopic traffic simulation. Paper 04-2414, Transportation Research Board Annual Meeting, Washington, D.C., 2004.

28. E. Kwon and S. Pitt. Evaluation of emergency evacuation strategies for downtown event traffic using a dynamic network model. Paper 05-2164, Transportation Research Board Annual Meeting, Washington, D.C., 2005.

29. Y.-C. Chiu, P. Korada, and P.B. Mirchandani. Dynamic traffic management for evacuation. Paper 05-2369, Transportation Research Board Annual Meeting, Washington, D.C., 2005.

30. X. Chen and F.B. Zhan. Agent-based modeling and simulation of urban evacuation: Relative effectiveness of simultaneous and staged evacuation strategies. Paper 04-0329, Transportation Research Board Annual Meeting, Washington, D.C., 2004.

31. L.D. Han and F. Yuan. Evacuation modeling and operations using dynamic traffic assignment and most desirable destination approaches. Paper 05-2401, Transportation Research Board Annual Meeting, Washington, D.C., 2005.

32. P. Sattayhatewa and B. Ran. Developing a dynamic traffic management model for nuclear power plant evacuation. Paper, Transportation Research Board Annual Meeting, Washington, D.C., 2000.

33. B. Barrett, B. Ran, and R. Pillai. Developing a dynamic traffic management modeling framework for hurricane evacuation. Paper 00-1595, Transportation Research Board Annual Meeting, Washington, D.C., 2000.

34. R. Alsnih and P. Stopher. A review of the procedures associated with devising emergency evacuation plans. Working Paper ITS-WP-04-04, Institute of Transport Studies, The University of Sydney and Monash University, 2004. See www.itls.usyd.edu.au/publications/working_papers/wp2004/its_wp_04-04.pdf.

35. E. Cascetta and C. Cantarella. A day-to-day and within-day dynamic stochastic assignment model. *Transportation Research A*, 25A(5):277–291, 1991.

36. G. Theodoulou and B. Wolshon. Alternative methods to increase the effectiveness of freeway contraflow evacation. *TRR*, 1865:48–56, 2004.

37. E. Lim and B. Wolshon. Modeling and performance assessment of contraflow evacation termination points. *TRR*, 1922:118–128, 2005.

38. I.J. Farkas. `pedsim` source code, accessed 2008. See pedsim.elte.hu.

39. C. Gloor, P. Stucki, and K. Nagel. Hybrid techniques for pedestrian simulations. In *Proceedings of Swiss Transport Research Conference (STRC)*, Monte Verita, CH, 2004. See www.strc.ch.

40. C. Gawron. An iterative algorithm to determine the dynamic user equilibrium in a traffic simulation model. *International Journal of Modern Physics C*, 9(3):393–407, 1998.

41. P.M. Simon, J. Esser, and K. Nagel. Simple queueing model applied to the city of Portland. *International Journal of Modern Physics*, 10(5):941–960, 1999.

42. MATSIM www page. MultiAgent Transport SIMulation. http://matsim.org/, accessed 2008.

43. H. Taubenböck and A. Roth. A transferable and stable classification approach in various urban areas and various high resolution sensors. In *Urban Remote Sensing Joint Event*, Paris, 2007.

44. A. Schadschneider, W. Klingsch, H. Klüpfel, T. Kretz, C. Rogsch, and A. Seyfried. Evacuation dynamics: Empirical results, modelling and applications. In B. Meyers, editor, *Encyclopedia of Complexity and System Science*. Springer, Berlin, to appear.

45. P.M. Simon and K. Nagel. Simple queueing model applied to the city of Portland. Paper 99-0861, Transportation Research Board Annual Meeting, Washington, D.C., 1999.

46. T. Kretz, A. Grünebohm, and M. Schreckenberg. Experimental study of pedestrian flow through a bottleneck. *Journal of Statistical Mechanics: Theory and Experiment*, P10014, 2006.

47. H.C. Muir, D.M. Bottomley, and C. Marrison. Effects of motivation and cabin configuration on emergency aircraft evacuation behavior and rates of egress. *International Journal of Aviation Psychology*, 6:57–77, 1996.

48. E. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1:269 – 271, 1959.

49. Q. Lu, B. George, and S. Shekhar. Capacity constrained routing algorithms for evacuation planning: A summary of results. *LNCS*, 3633:291–307, 2005.

50. BPS. *Kecamatan Dalam Angka - Subdistricts in Numbers*. Satistical bureau (BPS) Kota Padang, Padang, 2005.

51. N. Goseberg, A. Stahlmann, S. Schimmels, and T. Schlurmann. Highly-resolved numerical modeling of tsunami run-up and inundation scenarios in the city of Padang, West Sumatra. In *Interantional Conference on Coastal Engineering*, Hamburg, 2008. (Poster).

52. D. Helbing, I. Farkas, and T. Vicsek. Simulating dynamical features of escape panic. *Nature*, 407:487–490, 2000.