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: 6300 + 5 figures

ABSTRACT
The evacuation of whole cities or even regions is an important problem, as demonstrated by recent events such as 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. It is well-documented that 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 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 including evacuation simulations in the urban coastal hinterland for the city of Padang.

An important aspect is the amount of time that is needed for the evacuation. 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. With this simulation it should be possible to get an estimate of the evacuation process.

In this paper we describe the simulation framework for the pedestrian evacuation simulation from a technical point of view. We will not introduce the "Last-Mile Evacuation" project in detail. The interested reader is referred to *(2, 3)* for detailed information about the project, project partners and their particular work packages.

RELATED WORK

Disaster and evacuation planning has become an important topic in science and politics. In principle there are two different situations: evacuation of buildings, ships and airplanes or the like on the one hand, or evacuation of whole cities or even regions on the other hand. The former involves normally the evacuation of pedestrians, where the latter is rather associated with the evacuation by car.

Corresponding to the two different types of problems, there are two different basic approaches for simulating the traffic flow:

[1] Vertical evacuation means that it is planned to build quake and tsunami proof shelters, where the evacuees can flee to.

(1) Methods of dynamic traffic assignment (DTA) have been applied to evacuation simulation on the city or regional scale. Some examples are: MITSIM *(4)*, DYNASMART *(5)* or VISSIM *(6)*. The DTA approach is based on the analogy between traffic and hydrodynamic characteristics of fluids. That means DTA is a macroscopic approach and reduces the problem of evacuation dynamics to a well known physical problem. On state of the art hardware it is possible to handle even large-scale scenarios with this approach. – However, in DTA it is not straightforward to deal with the inhomogeneity of a population. For this, a microscopic simulation is needed, where all people are simulated as individuals.

(2) Microscopic simulations are often based on Cellular Automata (CA) *(7, 8, 9)*. In CA models each evacuee is designed as an individual; therefore it is possible to simulate population structures where people have different speeds or ranges, or more complex behavior. The modeling of complex behavior in evacuation simulation has become important in recent years. People could, for example, ignore warnings, or might not choose the nearest emergency exit, furthermore people tend to follow others (herd behavior) *(10, 11)*. Agent oriented research groups have modeled such behavior *(12, 13)*. In general it is expected that complex behavior leads to longer evacuation times, consequently a simulation that ignores such behavior patterns is probably optimistic.

The aim of this approach is to develop a simulation framework for large-scale scenarios, e.g. for large cities with a population of hundreds of thousands. A standard CA-based approach is not applicable here, because the area of those cities could be several hundred square kilometers. In this case a CA-model would consist of more than 10^9 cells, leading to rather long computing times.

In contrast, a DTA approach, as pointed out earlier, is not able to handle complex individual behavior. Accordingly, existing DTA based methods have been used to find general solutions for evacuation scenarios. In some examples *(4, 14)* the optimal distribution, in terms of minimizing the evacuation time, of the evacuees on the evacuation routes are determined. Other approaches try to estimate the human route choice behavior by applying discrete choice models *(15)*. However, this existing approaches lack on the capability of modeling individual behavior, like personal evacuation plans or social behavior.

One possible approach to deal with such large-scale scenarios but to retain persons as individual agents is based up on a modified queuing model *(16, 17, 18)*. 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. For example, the simulation of the whole (motor) traffic of Switzerland (approx. 5 million trips) takes less then 5 minutes for 24h real time *(19)*.

In this work the adaptation of the existing multi agent transportation simulation framework to large-scale pedestrian evacuation simulation is described. This work is motivated by the interest of finding feasible solutions for an evacuation of the Indonesian City of Padang in the case of a tsunami.

SIMULATION FRAMEWORK

The simulation framework is based on the MATSim framework for transport simulation *(20)*. Since MATSim is focused on simulation of motorized traffic, several adaptations 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.

Agent database

The agent database should reflect the socio-economic profile of the evacuation area. A general good starting point is to extract the needed information from census data. But, depending on the region other sources have to be taken into account (e.g. for vacation areas the expected distribution and capabilities of the vacationers.). However, census data often lack on the information needed to build activity chains. This is needed to estimate the distribution of the population as a function of time-of-day. One possibility to overcome this problem is by undertaking a survey. For Last-Mile there has been survey of 1000 households to get the needed information. The selection of the households was based on their socio-economic characteristics and the physical structure of their homes. The developed questionnaire also includes questions about the daily activities of the households. The survey results will allow derivation of the distribution of the population as a function of time. This is important since we want to develop evacuation scenarios for different times of the day. The preparation and implementation of the survey as well as analysis of the results will take several months. In the mean time a population distribution based on census data—assuming that all people are at home—is used for the evacuation simulation.

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. And the extraction has been done by using an object-oriented hierarchical classification approach *(21)*. After the raw street map was extracted we decomposed it into crossings and street segments. An exemplary picture of the so generated street network is shown in Fig. 1 a). However, to make the network usable for the physical evacuation simulation it had to be converted into a graph. This has been done by converting streets to links and crossings to nodes. 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 an evacuee wants to flee from one side of the street to the other side she has 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 got photographs of some of the medians and we could detect traversing options by manually looking over satellite imagery elsewhere. The so improved network

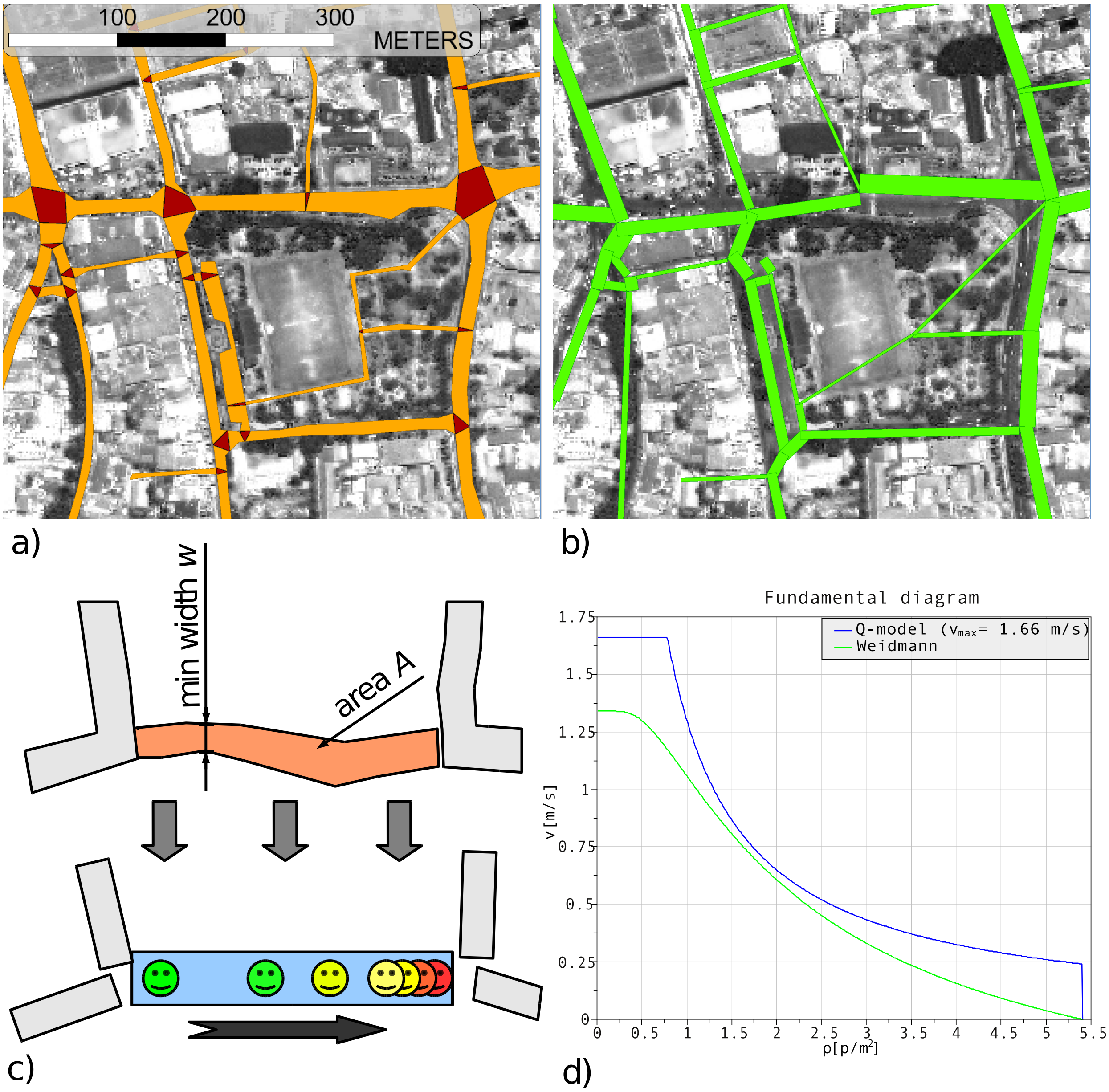


FIGURE 1 Street map extracted from Ikonos imagery (a) and the simulation network derived from this street map (b). Capacity restrictions in the queue model stem entirely from the links; in consequence, intersections are not modeled explicitly. Functioning of the queue model is shown in c) and its corresponding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

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 *(16)*. 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.

The parameters have been chosen to approximate Weidmann's fundamental diagram *(22)*.[2] He pointed out that the relation between density and velocity is adequately captured by the so-called Kladek-formula:

$$v_{F,hi}(D) = v_{F,hf} \times [1 - e^{-\gamma \times (\frac{1}{D} - \frac{1}{D_{max}})}]$$

With:

- $v_{F,hi}$ the velocity at a particular density $[m/s]$,
- $v_{F,hf}$ the velocity at free flow $[m/s]$,
- γ a free parameter $[persons/m^2]$,
- D the actual density $[persons/m^2]$ 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, 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 $v = \min[v_{max}, FC/D]$ *(17)*. The flow capacity FC is a free parameter that has to be chosen to fit the desired fundamental diagram. Even if a complete agreement is not possible, with $FC = 1.3\frac{p}{m*s}$ the flow dynamics produced by our queue model is not too far away from Weidmann's fundamental diagram (cf. Fig. 1 d)). Furthermore, Predtechenskii's and Milinskii's *(24)* empirical study supports a value of approx. $1.3\frac{p}{m*s}$ for the flow capacity.

Plans generation

Initial plans use the shortest path (according to free speed travel time) out of the evacuation area for all agents. Within the MATSim framework a shortest path router based on Dijkstra's shortest

[2] Newer studies *(23)* imply other fundamental diagrams then those from Weidmann. An adaptation of these values could, in consequence, become necessary in future.

path algorithm *(25)* 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[3].

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. *(26)*) 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.

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 needed time to evacuate). The scored plans remain in the agents' memory for further executions. For the learning procedure we applied two different learning strategies. 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, but using 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 by a logit model, where:

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

With:

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

In the long run this model is equivalent to the following probabilistic discrete choice model:

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

Where p_i is the probability for plan i to be selected and s_i its current score. In general both models are equivalent. 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.

A strategy selector decides for every agent which of the strategies (ReRoute or ChangeExpBeta) will be chosen. Each strategy is selected with a certain probability. These probabilities are assigned before the simulation starts, but they can be varied during the iterations.

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.

[3]For the initial evacuation plans the expected travel time is determined by free travel speed.

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. In most (but not all) evacuation situations, the Nash equilibrium leads to a shorter overall evacuation time than when everybody moves to the geographically nearest evacuation point. On the other hand, a Nash equilibrium means that nobody has an incentive to deviate. The Nash equilibrium in an evacuation situation can therefore be considered as a solution that could be reached by appropriate training.

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 the first scenario called: "evacuation at 3 am". That means we assume all inhabitants are at home. It is straight forward to derive the needed 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 *(27)*. 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. 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 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 convert to a steady state. The system would change so fast that the agents would not get a chance to exploit their knowledge

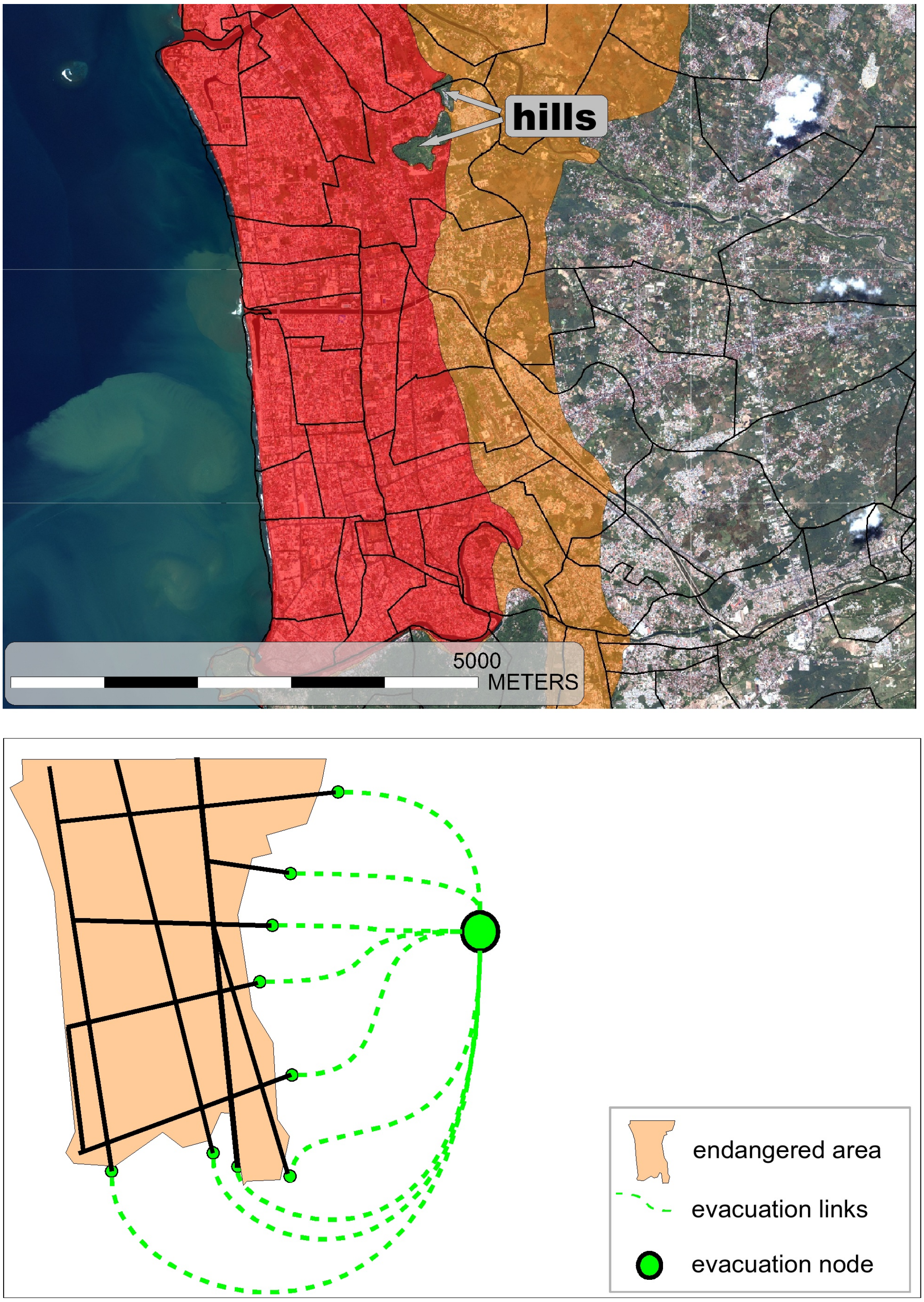


FIGURE 2 **The satellite imagery (top) of the city shows district borders and (highlighted) the evacuation area. The area with an elevation of up to $5\,m$ is colored in red, and $5\,m$ to $10\,m$ in orange. Bottom: A sketch of the evacuation network with the evacuation node as super sink and virtual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

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).

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 (bottom) 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. 4. 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 points to that no part of the city has significantly more bottlenecks, causing more congestion than other parts. The only exception is the area in the North-East around the hills (cf. Fig. 2). Many agents (approx. 100,000) evacuate to those hills causing congestions. These congestions could be reduced if the accessibility to those hills were improved. This is consistent with the overall observation made in relation to Fig. 3 (bottom) that the main impediment to faster evacuation is the cross-sectional capacity to the safe areas.

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 iterates 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 (bottom)) 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

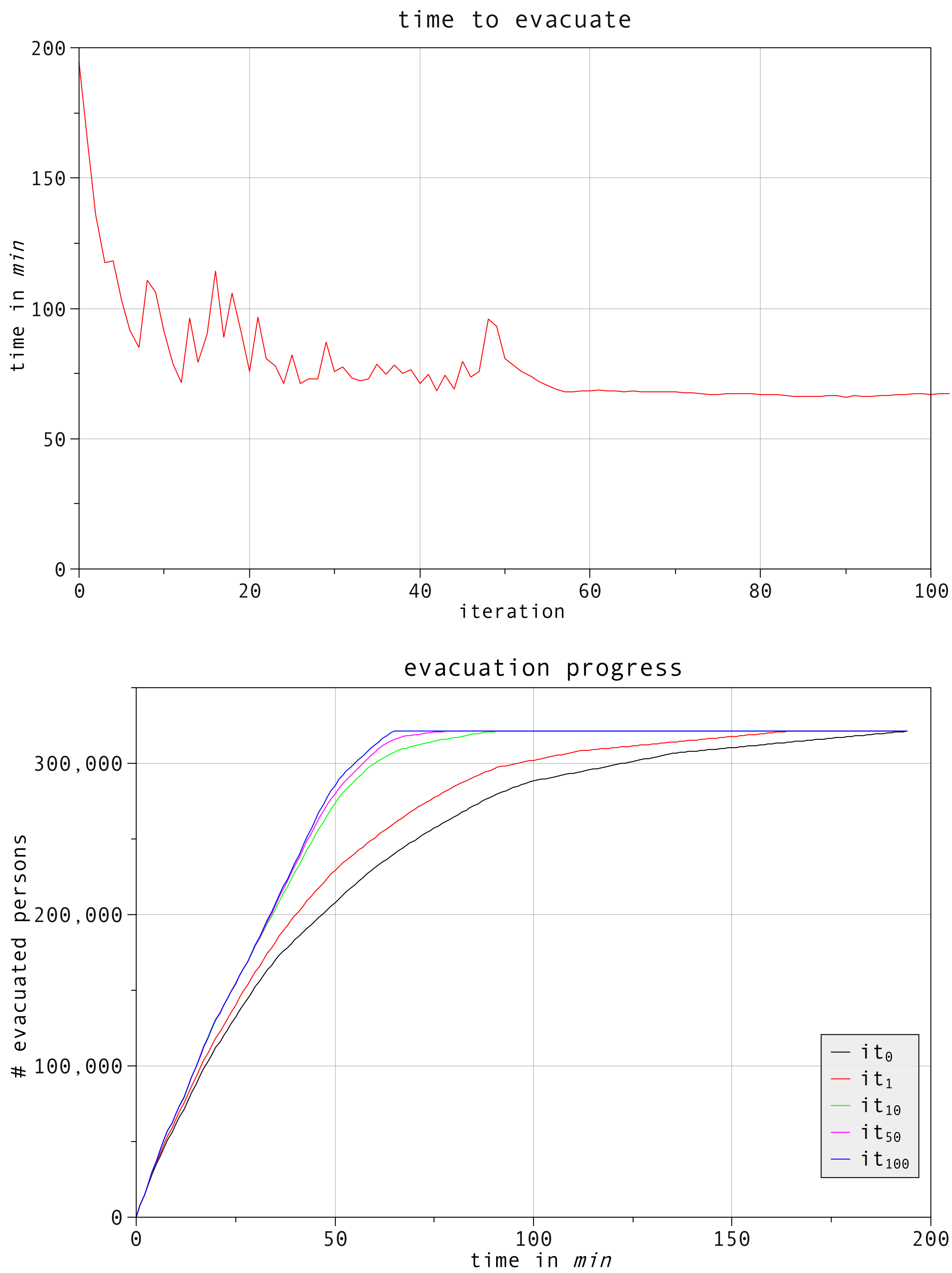


FIGURE 3 Evacuation time vs. iteration number (top) and evacuation progress showing the number of evacuated people as a function of time for various iteration numbers (bottom)

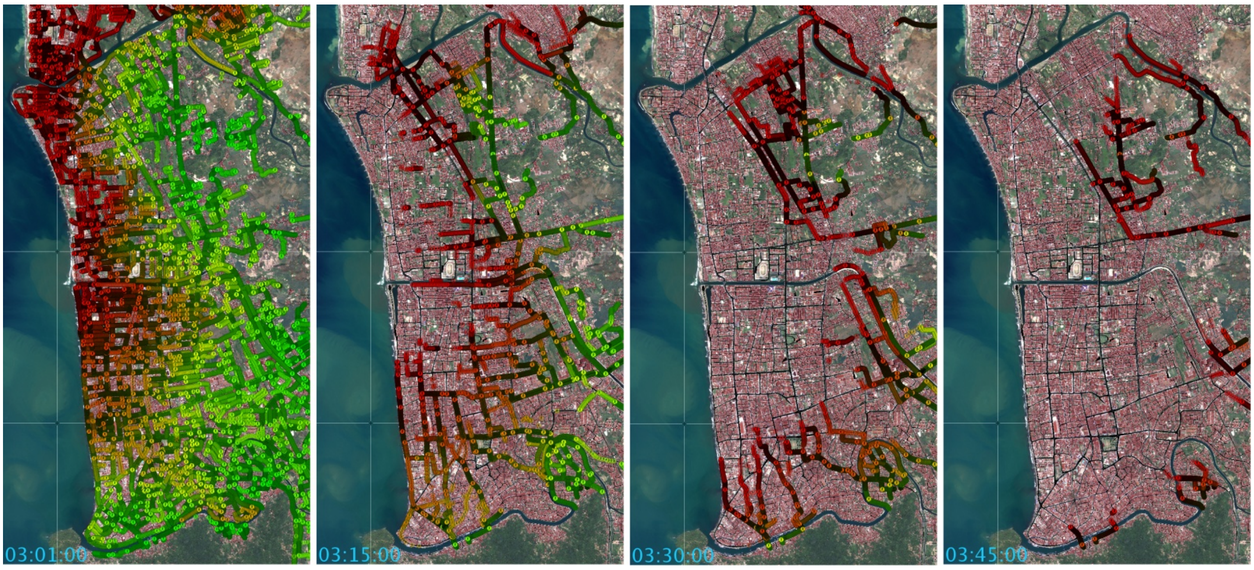


FIGURE 4 Visualizer snapshots with time dependent colorized agents after 100 iterations of learning. Satellite imagery by the German Aerospace Center, Oberpaffenhofen (2007)

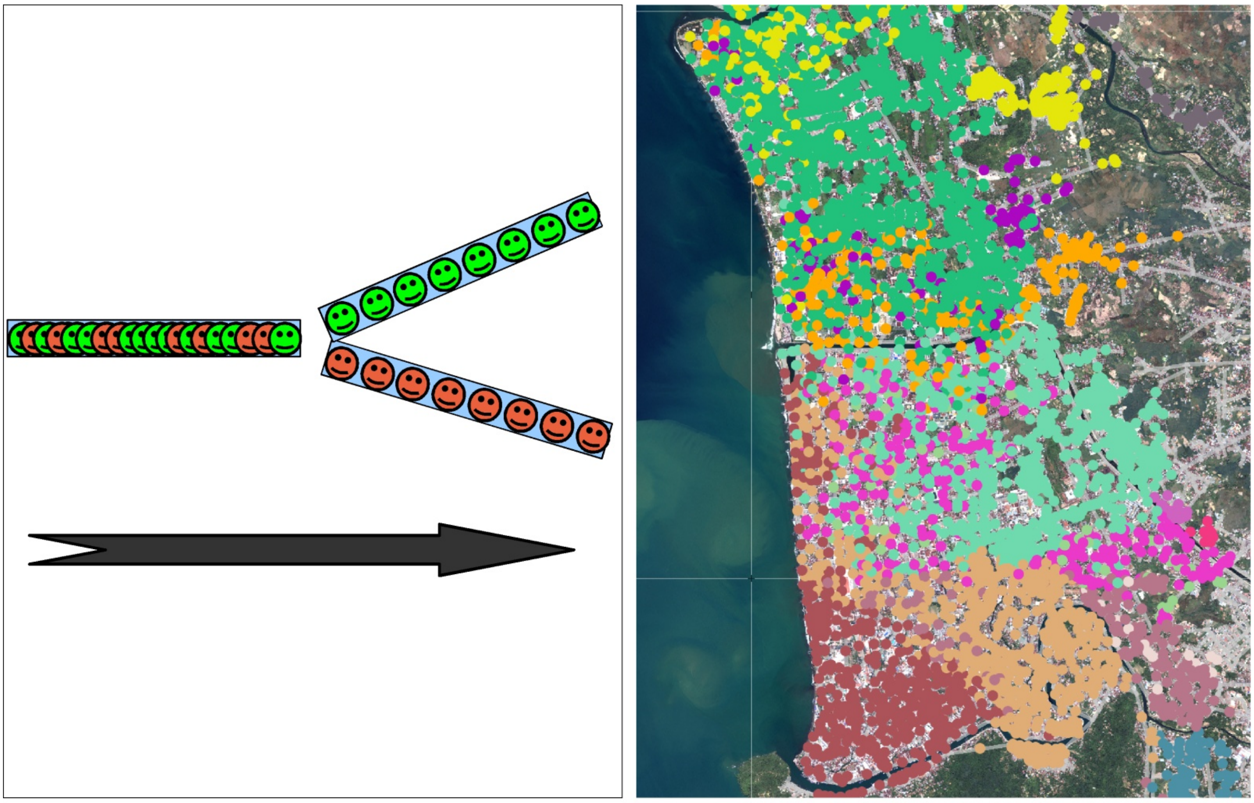
agents to do something that is not optimal for themselves. Compared to the system optimal solution, the Nash equilibrium solution has the advantage that one could attempt to reach it by training: Since the solution is constructed in a way that nobody could (unilaterally) gain by deviating from this solution, 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 *(10)*, 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 is that the found solution is not confluent. A solution would be confluent if at any node all the flow leaves along a single fixed link (see, e.g., portal.acm.org/citation.cfm?id=1007432). From such a solution one could directly derive evacuation



a) b)

FIGURE 5 Illustration of a diverging evacuation flow (a)) and home locations of the agents colorized depending on their evacuation destination after 100 iterations of learning (b)) Satellite imagery by the German Aerospace Center, Oberpaffenhofen (2007).

recommendations by applying a sign on every node (i.e. intersection) that points towards this link. In the case of an evacuation, every evacuee simply would have to follow these signs. But in our results, there are

- Links where evacuation flows diverge into multiple streams with possibly different destinations (Fig. 5 a)).
- Regions where agents from neighboring starting locations evacuate to different safety locations (Fig. 5 b)).

The former is a consequence of the situation that a high capacity link is followed by lower capacity links. In order to use the full capacity of the system, it is in this situation necessary that the stream diverges. The latter is a consequence of the Nash equilibrium approach: If all exits are similarly congested, it does not matter which exit you use. This is in part directly caused by the stream divergence problem: If there is a stream divergence, all locations upstream of the divergence point will, in equilibrium, have both options with the same cost. Probably for that reason, adding geographical distance to the cost function of the shortest path does not remove the effect, although it reduces it.

In a non-confluent situation, deriving a simple guidance scheme for evacuation is no longer possible. While it would still be possible to re-group neighboring starting locations to the same destination, the situation that people who are at some point on the same link may need to diverge later remains. The following approaches, none of them satisfactory, come to mind:

- For the Nash equilibrium solution, it is not necessary that the same people end up at the same destination every time; it is only necessary to maintain certain divergence fractions at the divergence points. Therefore, some kind of "shepard" at every divergence point who guides the appropriate fractions of people to the different exits might be an option.
- One could give people differently colored cards or tags, together with evacuation signs in different colors. People would have to follow the signs with "their" color.
- Signs at divergence points could be time dependent. Assuming a steady state inflow, a 1:2 divergence could be emulated by alternating, say, $3\ min$ of pointing to the first direction with $6\ min$ of pointing to the second direction. Variable direction signs for evacutions are in fact sometimes implemented in buildings (see, e.g., www.inotec-licht.de/DER.56.0.html), albeit for a different purpose (blockage of certain escape routes by a fire or similar).

Overall, we believe that it will be necessary to find much simpler schemes, based on simple rules of thumb and minimal infrastructure requirements.

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 *(10)* or *(28)*. Such models could still operate on networks *(29)*.

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 planed to derive the time dependent distribution of the population directly from the demand. The improvement of the behavioral model (e.g. herd behavior *(10)* modified for large-scale scenarios *(29)*) could also be a topic of future work.

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 would like to thank Daniel Dressler and Sandor Fekete for explanation of the confluent flow problem and Hubert

Klüpfel for the fruitful discussion about pedestrian evacuation simulations.

REFERENCES

1. D.H. Natawidjaja, K.S., 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*, Wuppertal, Germany, 2008.

4. 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.

5. 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.

6. 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.

7. K. Nagel and M. Schreckenberg. A cellular automaton model for freeway traffic. *Journal de Physique I France*, 2:2221–2229, 1992.

8. 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.

9. S.P. Hoogendoorn, P.H.L Bovy, and W. Daamen. Microscopic pedestrian wayfinding and dynamic modelling. In M. Schreckenberg and S. D. Sharma, editors, *Pedestrian and Evacation Dynamics*, Proceedings of the 1st international conference, Duisburg, 2001, pages 123–154. Springer, 2001.

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

11. C. Lieberman, D. Kurowski, M.R. Avila, L. Ricci, N. Thomas, C. Collyer, and B. Aguirre. Conceptual framework for simulating the pedestrian evacuation behavior from buildings. Paper 05-2297, Transportation Research Board Annual Meeting, Washington, D.C., 2005.

12. Y. Murakami, T. Ishida, T. Kawasoe, and R. Hishiyama. Scenario description for multi-agent simulation. In *Autonomous agents and multiagent systems (AAMAS'03)*, Melbourne, Australia, July 2003.

13. X. Pan, C.S. Han, K. Dauber, and K.H. Law. A multi-agent based framework for the simulation of human and social behavior during emergency evacuations. *AI and Society*, 22(2):113–132, 2007.

14. A.M. Afshar and A. Haghani. A heuristic framework for optimizing hurricane evacuation operations. Paper 08-2336, Transportation Research Board Annual Meeting, Washington, D.C., 2008.

15. A.J. Pel and M.C.J. Bliemer. Evacuation plan evaluation: Assessment of mandatory and voluntary vehicular evacuation schemes by means of an analytical dynamic traffic model. Paper, Transportation Research Board Annual Meeting, Washington, D.C., 2008.

16. 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.

17. 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.

18. 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.

19. B. Raney, N. Cetin, A. Völlmy, M. Vrtic, K. Axhausen, and K. Nagel. An agent-based microsimulation model of Swiss travel: First results. *Networks and Spatial Economics*, 3(1):23–41, 2003.

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

21. 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.

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

23. 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.

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

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

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

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

28. H. Klüpfel. The simulation of crowd dynamics at very large events – Calibration, empirical data, and validation. In P. Gattermann, N. Waldau, and M. Schreckenberg, editors, *Pedestrian and Evacuation Dynamics*, Proceedings of the 3rd international conference, Vienna. Springer, Berlin, 2006.

29. 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.