EVACUATION SIMULATION WITH LIMITED CAPACITY SINKS

An evolutionary approach to solve the shelter allocation and capacity assignment problem in a multi-agent evacuation simulation

Gunnar Flötteröd
Transport and Mobility Laboratory, EPFL - Ecole Polytechnique Fédérale de Lausanne, Lausanne, Switzerland
gunnar.floetteroed@epfl.ch

Gregor Lämmel
Transport Systems Planning and Transport Telematics, Berlin Institute of Technology, Berlin, Germany
laemmel@vsp.tu-berlin.de

Keywords: evacuation, shelter allocation, shelter capacity assignment, iterative learning, Nash equilibrium, system optimum, multi-agent simulation.

Abstract: We heuristically solve an evacuation problem with limited capacity shelters. An evolutionary learning algorithm is developed for the combined route- and shelter-assignment problem. It is complemented with a heuristic method for the fair minimization of shelter capacities. Different behavioral assumptions “fair” vs. “globally optimal”) are investigated. The proposed approaches are discussed in the context of a real-world tsunami evacuation problem.

1 INTRODUCTION

The evacuation of whole cities or even regions is a problem of substantial practical relevance, which is demonstrated by recent events such as the evacuation of Houston because of Hurricane Rita or the evacuation of coastal cities in the case of tsunamis. Important tools for the planning of organized reactions to such events are model-based simulation systems.

The general evacuation problem is to minimize the egress time of an endangered region or building by assigning a feasible escape route and destination to every evacuee. This problem is complex because of congestion effects that inevitably occur when many evacuees enter the transportation facilities (roads, hallways, stairways) at once.

In some evacuation scenarios, there exist secure areas of limited capacity within the evacuation zone, such as shelter buildings in tsunami prone areas. Concrete applications of mathematical programming techniques to the evacuee–shelter–allocation problem can be found in (Sherali et al., 1991; Peng, 2006).

This paper deploys a detailed microsimulation for the representation, analysis, and optimization of pedestrian evacuation dynamics for a tsunami situation in a large coastal metropolitan area. Building on existing routing strategies (Lämmel and Flötteröd, 2009), it provides new solutions to (i) the combined route and shelter assignment problem and (ii) the shelter capacity assignment problem, considering both ”fair” and ”optimal” assignment rules.

The added value of the agent-based approach is its natural representation of individual travelers as software agents that interact in a simulated version of the real world (a virtual environment). This is an advantage over analytical models in that it allows (at least technically) for a much higher model resolution. However, it comes at the price of greater difficulties in the mathematical treatment of the problem. The optimization results presented in this article are therefore only of an approximate nature.

2 PROBLEM STATEMENT

In a first step, we investigate different strategies to assign routes and destinations (shelters) to evacuees. In a second step, we identify optimal dimensions of the shelters. Overall, we consider two different objectives:

Fairness. No evacuee will agree to take an obvious detour or to select an obviously faraway shelter instead of a nearby one. This requires to identify route and shelter assignments that are fair in that no evacuee can obviously gain by switching to a different route or shelter. It corresponds to a Nash equilibrium of all evacuation strategies in the population.

Efficiency. It is desirable to evacuate the system as quickly as possible. While a Nash strategy has the obvious and important advantage of general acceptance, it may be suboptimal in this regard because some evacuees may do great damage to others by blocking their ways/shelters. We therefore identify approximations of optimal evacuation strategies as benchmarks to which fair solutions can be compared.

An important topic for future research is to combine both approaches into evacuation strategies that are more efficient than Nash equilibria without introducing obvious levels of unfairness.

2.1 Simulation framework

We model the urban evacuation region and the population of evacuees with a multi-agent simulation, where every single person is individually represented. For this purpose, the MATSim simulation framework is adopted (MATSim, 2010). MATSim is designed for the computation of transport equilibria, and hence it can be immediately deployed for the computation of Nash evacuation strategies. Some adjustments are necessary for approximately optimal strategies.

MATSim allows for adjustments in the different choice dimensions of a simulated traveler through *modules*, where, typically, one module is responsible for one choice dimension. In our application, this requires to specify four modules: (1) Nash route choice, (2) Nash destination (shelter) choice, (3) optimal route choice, (4) optimal shelter choice.

MATSim computes approximate Nash equilibria by iterating best-response behavior: in every iteration, a fraction of the travelers recalculates a route or a destination based on what would have been best in the previous iteration, assuming that the behavior of all other agents stays unchanged. After this replanning, the resulting *plans* of all travelers are simultaneously executed in the mobility simulation and new performance measures are computed. This process is repeated many times. Once it stabilizes, no agent can substantially improve through a route or destination replanning, and an approximate Nash equilibrium is obtained.

An alternative assignment logic is to not compute best responses in every iteration but *cooperative* behavior that improves the situation of the population as a whole. The more involved realization of such behavior follows essentially the same simulation logic as the Nash assignment, but with a modified cost function being presented to the agents.

2.2 Network modeling

The evacuation network consists of a set of nodes that are connected by a set of directed links. Sources (origins) as well as sinks (destinations, shelters) are associated with respective node subsets.

Every destination d has a capacity c_d that represents the maximum number of evacuees it can shelter. Destination nodes may also be located at the boundary of the endangered area, in which case they do not provide a limited shelter but access to a safe region, which is modeled by assigning them an unlimited capacity.

We consider a pedestrian simulation scenario on a road network, where the intersections correspond to nodes and the street segments connecting the intersections are modeled through links. Basic pedestrian traffic flow dynamics are captured through a limited number of link parameters: outflow capacity (maximum number of pedestrians the link can emit per time unit), space capacity (maximum number of pedestrians in the link), and maximum velocity (in uncongested conditions). Note that this formalism can be immediately transferred to vehicular evacuation problems (Cetin et al., 2003).

3 ROUTE ASSIGNMENT

Given that every evacuee $n = 1 \ldots N$ is assigned to a shelter $d(n)$, the route assignment problem is to find a feasible and in some sense best route from that evacuee's origin $s(n)$ to her shelter.

3.1 Nash equilibrium assignment

In the given context, a Nash equilibrium describes a situation where no evacuee can gain by unilaterally deviating from her current route (Nash, 1951). Since a Nash equilibrium means that nobody has an incentive to make a change, it can be considered as a socially acceptable and hence implementable evacuation strategy.

In a multi-agent (evacuation) simulation, the solution can be moved towards a Nash equilibrium through iterative learning (Gawron, 1998). As described above, such an algorithm starts with a given (routing) strategy for every agent, and then adjusts this strategy through some trial and error mechanism. In the given evacuation context, strategies are only evaluated based on their travel times.

Formally, the real-valued time is discretized into K segments ("bins"), which are indexed by $k = 0 \ldots K-1$. The time-dependent link travel time when

Algorithm 1 Nash equilibrium routing

1. initialize $\tau_a(k)$ with the free-flow travel time for all links a and time steps k
2. repeat for many iterations:
   (a) recalculate routes based on time-dependent link costs $\tau_a(k)$
   (b) simulate agent movements, obtain new $\tau_a(k)$ for all a and k

entering link a in time step k is denoted by $\tau_a(k)$. Alg. 1 drafts the Nash-equilibrium routing logic.

3.2 System optimal assignment

A system optimal routing solution minimizes the total travel time in the system. Classical solutions to this problem apply mathematical programming techniques, which are based on the theory of dynamic network flows. The foundations of these techniques have been laid in (Ford and Fulkerson, 1962), and dynamic flow models have been applied to evacuation problems from the early 1980s on (see, e.g., (Chalmet et al., 1982)).

In the multi-agent domain, an approximate system optimum (SO) can be found through an iterative learning approach that is closely related to the simulation of a Nash equilibrium as described above (Lämmel and Flötteröd, 2009).

The only difference to the Nash routing logic given in Algorithm 1 is that the travel time based on which agents evaluate their routes is replaced by the marginal travel time (Peeta and Mahmassani, 1995). The marginal travel time of a link is the amount by which the total system travel time changes if one additional traveler enters that link. It is the sum of the cost experienced by the added traveler ($\tau_a(k)$) and the cost imposed on all other travelers. The latter is denoted as the time-dependent "social cost".

Letting each evacuee *individually* minimize her marginal travel time implicitly enforces a *cooperative* behavior that also minimizes the total system travel time. This maximizes the number of evacuees who have reached their destinations in each time step, which in turn minimizes the egress time (Jarvis and Ratliff, 1982).

4 SHELTER ASSIGNMENT

The shelter assignment problem is to identify, for each evacuee, if this evacuee should access a shelter or not and, given that a shelter is accessed, to decide which shelter. Again, both a Nash and an SO approach are possible.

Algorithm 2 Nash shelter allocation

1. initialize routes and destinations for all agents
2. repeat many times
   (a) load all agents on the network
   (b) extract link travel times
   (c) for every agent $n = 1 \ldots N$, do with P_{replan}:
   - with $P_{reroute}$, compute a new route from $s(n)$ to $d(n)$.
   - with P_{move},
     i. randomly select a non-full shelter d'
     ii. compute the benefit of a move: $\delta = c_{s(n)d(n)} - c_{s(n)d'}$
     iii. if $\delta > 0$, assign d' as the new destination to n and re-route n
   - with P_{switch},
     i. randomly select n' from $\{1, \ldots, N\}$
     ii. compute the minimum benefit of a switch: $\delta = min(c_{s(n)d(n)} - c_{s(n)d(n')}, c_{s(n')d(n')} - c_{s(n')d(n)})$
     iii. if $\delta > 0$, then switch the destinations of n and n' and re-route both agents

4.1 Nash shelter assignment

We extend the individual best-response logic to a pairwise best response, where for every agent n that replans its shelter a "switching partner" n' in another shelter is randomly selected, and both agents switch their shelters if and only if *both* benefit from this switch. This decision is made based on the expected travel time of a best-response re-routing to each new destination; the resulting routes are also adopted in the case of an accomplished switch.

The iterative simulation conducts a shelter switch with a certain probability P_{switch}, and it also maintains the option of a plain route recomputation with $P_{reroute}$. Algorithm 2 defines the details of this logic.

The origin of agent n is denoted by $s(n)$ and its destination by $d(n)$. The cost $c_{s(n)d(n)}$ corresponds to the travel time from $s(n)$ to $d(n)$. Step 1., 2.(a), 2.(b) and the first step of 2.(c) are symmetric to the routing logic of Algorithm 1. An agent moves with probability $P_{replan} * P_{move}$ to a non-full shelter if it would benefit from that move. With probability $P_{replan} * P_{switch}$, two agents switch their shelters if both of them would benefit.

4.2 SO shelter assignment

Technically, the SO shelter assignment does not function differently from the Nash shelter assignment, only that two agents now "agree" to switch their shelters if this reduces the total travel time in the system. To decide this, the expected change in *marginal* travel times is evaluated before and after the switch.

Algorithm 2 is still applicable with the following modifications: $c_{s(n)d(n)}$ now represents agent n's marginal travel time, cf. Section 3.2. A switch is only performed if both agents would benefit from it. Therefore, the switching benefit δ of Step 2. (c) ii. needs to be computed as $c_{s(n)d(n)} + c_{s(n')d(n')} - c_{s(n)d(n')} - c_{s(n')d(n)}$.

5 SHELTER CAPACITY ASSIGNMENT

The shelter capacity assignment problem is to minimize the total shelter capacity subject to the constraint that no evacuee takes damage from being neither able to reach the safe area nor to enter a shelter because of lacking capacity. That is, a configuration is required where only those evacuees are assigned to shelters who would not make it to the safe region otherwise.

5.1 Shelter capacity assignment subject to Nash constraints

We base our approach on the Nash simulation logic of Algorithm 2. If there is more shelter capacity than strictly needed, there are likely to be agents in the shelters that could also make it to the safe region (because it can be assumed that for many such agents the shelter still is closer than the safe area). It is not feasible to *ex post* remove these agents from the shelters and to constrain the shelter capacities accordingly because this would change the travel times and hence the survival chances of the needy agents. The shelter capacities therefore need to be gradually adjusted during the iterations.

This effect is achieved by evaluating, in every iteration, the space occupied in every single shelter by agents that would also have made it to the safe region. If this surplus is vanishing, the shelter is urgently needed, and its capacity is increased by a relative amount (say, 5 percent). If, on the other hand, this surplus is substantial, the shelter is too large, and it is shrunk by a relative amount (say, again, 5 percent) of its surplus capacity.

Algorithm 3 Heuristic Nash shelter allocation and capacity assignment

1. initialize routes and destinations for all agents
2. repeat many times
   (a) load all agents on the network
   (b) extract link travel times
   (c) for every agent $n = 1 \dots N$, replan with P_{replan} its route/destination; give strict preference to needy agents in shelter assignment
   (d) for every shelter $d = 1 \dots D$, do:
      - $o(d) = c(d) - \sum_{n=1}^{N} x_{nd} + \sum_{n=1}^{N} \hat{y}_n$
      - if $o(d) > 0$, decrease $c(d)$ by $min(o(d), q * c(d))$ and re-route non-needy agents to super-shelter if necessary; else increase $c(d)$ by $q * c(d))$

This mechanism, in combination with a strict preference for needy agents in the shelter allocation, eventually leads to a configuration where all available shelter capacity is allocated to needy agents, given otherwise fair Nash equilibrium conditions. Algorithm 3 gives an overview.

The variable x_{nd} indicates the allocation of agent n to shelter d, i.e., $x_{nd} = 1 \Leftrightarrow d(n) = d$. y_n indicates if agent n has enough time to evacuate to the super shelter. If agent n has enough time to evacuate to the super shelter $y_n = 1$ and $y_n = 0$ otherwise. $\hat{y}_n$ is the estimated value of y_n based on the experienced travel costs from previous iteration. q denotes the relative amount by which the capacity of a shelter can change at most. The super-shelter represents the entire safe area.

5.2 Shelter capacity assignment subject to SO constraints

The only change when going from a shelter capacity assignment subject to Nash constraints to one subject to SO constraints is that the route choice and shelter switching behavior of all re-planning agents is conducted according to the SO logic described in Subsections 3.2 and 4.2. The conditions for shelter capacity decreases and increases in the SO case are the same as for the Nash case given in in Algorithm 3.

6 EXPERIMENTS

The Indonesian city of Padang, located at the West Coast of Sumatra Island, is exposed to earth quake triggered tsunamis. The evacuation street network consist of approx. 12 500 unidirectional links and 4 500 nodes. There are in total 224 798 evacuees. This corresponds to the number of persons living in the evacuation area. 42 hypothetical shelter buildings with a total capacity of roughly 31 500 evacuees are placed in the network. A sketch of the network including the shelters is given in Figure 1. The gray-shaded area has to be evacuated.

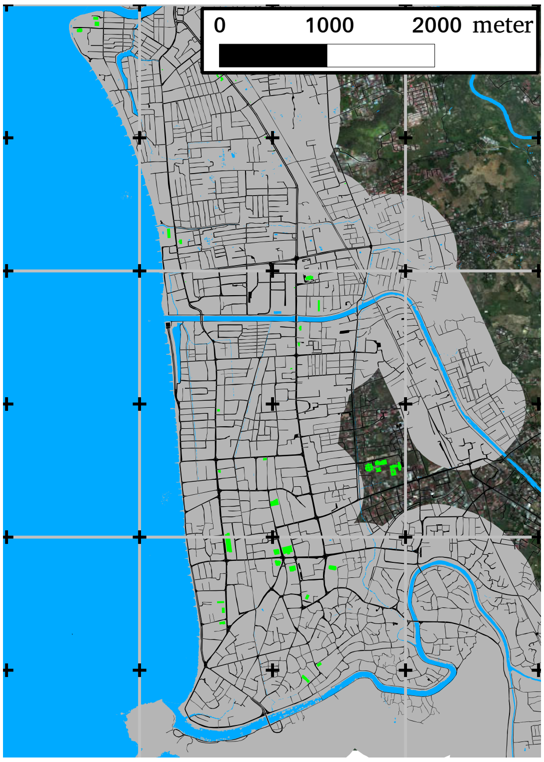

Figure 1: Map of the evacuation area.

We conduct four different simulations.

- *Run 1* implements the Nash equilibrium routing and shelter allocation.
- *Run 2* implements the Nash equilibrium routing and shelter allocation and the shelter capacity assignment.
- *Run 3* implements the SO routing and shelter allocation.
- *Run 4* implements the SO routing and shelter allocation and the shelter capacity assignment.

These runs are performed with a 10% sample of the population. The shelter capacities and network flow parameters are accordingly scaled down to 10%. This procedure saves computing time while staying reasonably realistic. With this setup, a simulation with 2000 iterations takes between 05:30 h (*Run 1*) and 10:30 h (*Run 4*) on a 2.66 GHz CPU running 64 bit Java on Linux. The memory consumption is below 3 GB in all experiments.

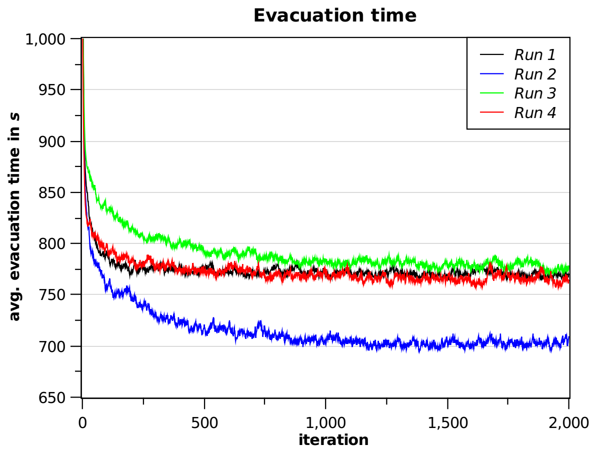

Figure 2: Average evacuation time per agent versus iteration number.

At the beginning of all runs, the agents are assigned randomly to the shelters. Figure 2 shows the average evacuation time per agent over the iteration number. *Run 1* shows that a Nash shelter assignment leads to considerably better evacuation times than a random shelter assignment. Adding the shelter capacity assignment reduces the average evacuation times further (*Run 2*). However, this comes at the cost of a drastic increase in shelter capacities, which is shown in Figure 3.

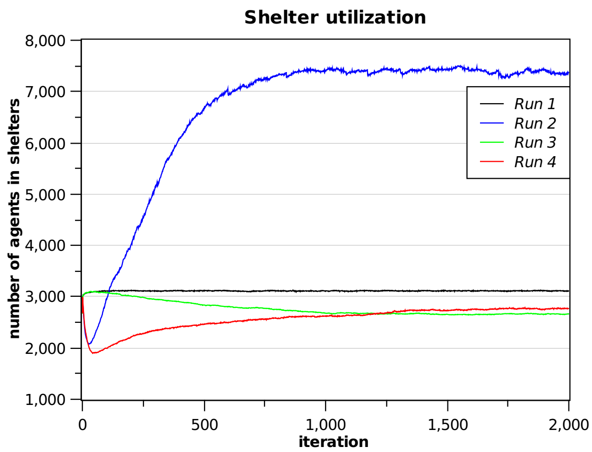

Figure 3: Change of the shelters total utilization over the learning iterations.

The SO in *Run 3* and *Run 4* is realized by adding social costs to the travel time on each link.

Figure 2 reveals that the SO experiments *Run 3* and *Run 4* yield higher travel times than the Nash runs. However, the SO runs result in a substantially lower shelter utilization, cf. Figure 3. The higher travel times in the SO runs are likely to result from those travelers that are kept out of the shelters and hence have to take longer routes to the safe city boundaries.

In theory, at least *Run 3* should outperform *Run 1*. The fact that this is not the case is due to the approximate nature of the deployed algorithms, which clearly need further attention.

7 DISCUSSION AND SUMMARY

The safest strategy for a tsunami-threatened city like Padang would arguably be to build a tsunami proof shelter for every single person. However, this would exceed any financial resources. The relevant question thus is to identify which shelters are actually needed by persons who cannot be evacuated out of the city in time. The proposed algorithms help to identify these shelters and their capacities under different behavioral assumptions.

An important result of the simulation studies is that the required shelter capacity in the Nash equilibrium case is much higher than in the SO case. If one wanted to achieve the highest benefits with the least effort, one could implement the shelter configuration of *Run 4* and distribute some kind of tickets to the people that are allowed to enter a shelter. Those tickets could be preferably handed out to the most vulnerable people like the elderly or pregnant women.

Summarizing, a learning framework that approximately solves the shelter allocation and capacity optimization problem is presented and tested on a real-world scenario. The learning framework can be configured either to attain an approximate Nash equilibrium, where individual travel times are minimized non-cooperatively, or an approximate system optimum, where the global travel time is minimized in a cooperative manner.

The experiments show that both approaches yield reasonable results and that substantial savings in shelter capacity are possible if the evacuation behavior can be influenced to deviate from a perfectly fair Nash equilibrium towards a system optimum.

In future work, the proposed algorithms could be extended to also identify appropriate shelter locations. This could be realized by starting with a high number of shelter buildings, followed by a successive removal of underutilized shelters. Another interesting topic for further research would be to investigate the relative effect of capacity improvements in the transportation system when compared to investments in increased shelter capacities.

ACKNOWLEDGEMENTS

This project was funded in part by the German Ministry for Education and Research (BMBF) under grants 03G0666E ("last mile") and 03NAPI4 ("Advest").

REFERENCES

Cetin, N., Burri, A., and Nagel, K. (2003). A large-scale agent-based traffic microsimulation based on queue model. In *Proceedings of the 3rd Swiss Transport Research Conference*, Monte Verita/Ascona, Switzerland.

Chalmet, L., Francis, R., and Saunders, P. (1982). Network models for building evacuation. *Management Science*, 28:86–105.

Ford, L. and Fulkerson, D. (1962). *Flows in Networks*. Princeton University Press.

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

Jarvis, J. and Ratliff, H. (1982). Some equivalent objectives for dynamic network flow problems. *Management Science*, 28:106–108.

Lämmel, G. and Flötteröd, G. (2009). Towards system optimum: Finding optimal routing strategies in time-tependent networks for large-scale evacuation problems. volume 5803 of *LNCS (LNAI)*, pages 532–539, Berlin Heidelberg. Springer.

MATSim (accessed 2010). MATSim web site. http://www.matsim.org.

Nash, J. (1951). Non-cooperative games. *Annals of Mathematics*, 54(2):286–295.

Peeta, S. and Mahmassani, H. (1995). System optimal and user equilibrium time-dependent traffic assignment in congested networks. *Annals of Operations Research*, 60:81–113.

Peng, L. (2006). *A DYNAMIC NETWORK FLOW OPTIMIZATION FOR LARGE-SCALE EMERGENCY EVACUATION*. PhD thesis, City University of Hong Kong.

Sherali, H., Carter, T., and Hobeika, A. (1991). A location-allocation model and algorithm for evacuation planning under hurricane/flood conditions. *Transportation Research Part B: Methodological*, 25(6):439 – 452.