A Comprehensive Evacuation Planning Model and Genetic Solution Algorithm[*]

Marc Goerigk[†1], Kaouthar Deghdak[2], and Philipp Heßler[1]

[1]Fachbereich Mathematik, Technische Universität Kaiserslautern, Germany

[2]Laboratoire d'Informatique, Ecole Polytechnique de l'Université de Tours, France

Abstract

We consider the problem of evacuating an urban area caused by a natural or man-made disaster. There are several planning aspects that need to be considered in such a scenario, which are usually considered separately, due to their computational complexity. These aspects include: Which shelters are used to accommodate evacuees? How to schedule public transport for transit-dependent evacuees? And how do public and individual traffic interact? Furthermore, besides evacuation time, also the risk of the evacuation needs to be considered.

We propose a macroscopic multi-criteria optimization model that includes all of these questions simultaneously. As a mixed-integer programming formulation cannot handle instances of real-world size, we develop a genetic algorithm of NSGA-II type that is able to generate feasible solutions of good quality in reasonable computation times.

We extend the applicability of these methods by also considering how to aggregate instance data, and how to generate solutions for the original instance starting from a reduced solution.

In computational experiments using real-world data modelling the cities of Nice in France and Kaiserslautern in Germany, we demonstrate the effectiveness of our approach and compare the trade-off between different levels of data aggregation.

Keywords: disaster management; evacuation planning; location-routing; multi-criteria genetic algorithms; data aggregation

[*]Partially supported within the project DSS_Evac_Logistic, by the Federal Ministry of Education and Research Germany as FKZ 13N12229, and by the French National Research Agency as ANR-11-SECU-002-01 (CSOSG 2011).

[†]Corresponding author. Email: goerigk@mathematik.uni-kl.de

1. Introduction

1.1. Motivation

Due to an increasing number of natural and man-made catastrophes, there is an increasing demand for evacuation planning in urban areas. Whenever such a catastrophe takes place, the persons in charge of the evacuation process need to consider two main questions: Where do we evacuate to, and how do we evacuate there? In recent years, methods from operations research have been employed to help answering these two problems. Recent special issues [She07, DGVW11] underline the increasing interest from the research community.

Several subproblems arise. These include the transport of transit-dependent evacuees, the provisioning of evacuated people, traffic control avoiding congestion, and many more. From a mathematical point of view, an integrated problem approach considering several such questions simultaneously can potentially find solutions that have higher quality than those found by a sequential problem solving approach. However, major drawbacks are increased computation times, and reduced acceptability of solutions generated this way for practitioners, who are justifiably hesitant to accept a "black box" solution.

In this paper, we propose an integrated model that considers the location choice of shelters, bus routing for public transport, and routing for individual traffic simultaneously, considering both evacuation time and evacuation risk. To find solutions with acceptable computation times, we develop a genetic algorithm to solve the model heuristically, and discuss data aggregation techniques to further simplify the solution process.

The proposed model is part of a Decision Support System (DSS) in which the evacuation planner can interactively consider several scenarios, and can directly compare and choose between suggested solutions. Its usage is thus intended in the *preparedness* phase of disaster management [AI06].

1.2. Related Literature

For general surveys on evacuation planning and disaster management, we refer to [HT01, AI06, YAM08]. In the following, we briefly summarize research on the fields that are most related to the model proposed in this work.

The integration of public transportation and shelter location is an active research area [PP14]. One of the first papers on location decisions in evacuation planning is [SCH91], proposing a model in which the shelter locations and routes of the evacuees are chosen such that the optimal evacuation time is achieved. In contrast to their system optimal approach, [KCY05] models the evacuation problem using bi-level programming inspired by Stackelberg games. They solve the problem heuristically with a genetic algorithm. Similarly, also [LNXD12] consider a bi-level location-routing problem with equilibrium constraints.

Apart from the location of the shelters, the choice of routes on which people are evacuated is a crucial aspect, which decides whether an evacuation plan can succeed or not. Several reviews tackle the routing evacuation problem. To model these problems, network flow approaches or traffic assignment approaches are used. Early models focus on building evacuation, as for example [CFS82]. For urban area evacuation, [Yam96] models city evacuation as a minimum cost flow problem to assign the pedestrian evacuees to shelters, while the routes are calculated by solving shortest path problems. In [LGS05], a static network is used with time dependent node capacity as well as time dependent arc capacity to define an evacuation plan, i.e., to define routes and timetables to minimize the total evacuation time.

Recently, [Bis11] has introduced a model for bus-based evacuation planning. The choice of buses as a transportation mean has been motivated by the fact that car-based evacuation is logistically complex, expensive, produced unacceptable levels of congestion, and is more dangerous than bus-based evacuation. To solve the bus evacuation problem, Bish proposes a mixed integer program and two heuristics. For a variant of this problem, [GGH13] propose several Branch-and-Bound algorithms. Robust bus evacuation models have been considered in [GG14, GDT13].

A recent study [Bre12] introduced the multiple commodity evacuation problem using buses and vehicles. A mixed integer program is proposed, where the number of lanes in each arc is represented by integer variables. The objective is to minimize a weighted linear combination of the flows of the commodities arriving at their corresponding destinations and the total number of emergency lanes. This objective respects that in each intersection there is a penalty if an emergency lane is assigned and that there are also a priority weights for each kind of commodity.

Regarding the multi-criteria approach used in this paper, genetic algorithms are a popular heuristic solution method (see, e.g., the surveys [KCS06] and [CVVL02]). Especially the NSGA-II approach as introduced in [DPAM02] shows good performance in many applications. The advantage of population-based methods in general is that it is possible to generate a set of diverse solutions in only one single program run. Thus, there are many applications from evacuation planning that make use of genetic algorithms for multi-criteria problems, see, e.g. [SMT09, TCH07, GPKM10, AAW10]. In [YK07], an ant colony approach is proposed for dispatching commodities to distribution centers, and to evacuate wounded people simultaneously.

A frequent problem that occurs when real world problems are tackled using optimization models is the size of the input data. Often this data has to be aggregated in order to achieve feasible running times. Among the first to consider aggregation in location decision problems were Kuhn [HK78] and Zipkin [Zip80, Zip82]. Since then there have been several publications dealing with aggregation, especially in location theory. We refer to the survey papers on aggregation in optimization [RPWE91] and on aggregation in location analysis [FLRT08] for further details on this topic.

1.3. Contributions and Outlook

We propose an integrated model which consists of location aspects (choice of shelter locations), routing aspects (bus and individual traffic), as well as risk aspects on the chosen routes.

Our model is a multiple commodity problem using buses and vehicles. In contrast to the models presented by [Bre12], we solve a multi-criteria problem, which minimizes the total evacuation time, the risk exposure of the evacuees, and the number of shelters that are used.

We propose a metaheuristic solution approach based on NSGA-II. Furthermore, we directly consider the impact of instance aggregation on the solution approach. Using real-world evacuation instances of the cities of Nice in France and Kaiserslautern in Germany, we evaluate the effectiveness of the proposed approach for different degrees of data aggregation.

The remainder of this paper is structured as follows: In Section 2, the proposed model is described and an integer program formulation is given. In Section 3, we introduce a heuristic solution method based on NSGA-II. The impact of data aggregation is discussed in Section 4. We experimentally evaluate our solution methods in Section 5. Finally, Section 6 concludes the paper.

2. Problem Description

2.1. Evacuation Scenario

In the case of a natural or man-made catastrophe there is a need to transport people from the affected disaster area to safe places (e.g., gymnasiums or stadiums). We call these safe places shelter locations. People can reach these shelter locations in several ways, e.g., by car, by bus, by train, or on foot. In a large-scale evacuation, the shelter locations have to supply the evacuees with food, water, a place to sleep, medicine, etc. It might be necessary to give this supply for several hours up to several days. During the evacuation it is better to use roads that pass through safe ground and not through the endangered zone. The evacuees' aim is to reach a shelter as fast as possible. The authorities that carry out the evacuation are also interested in the cost of the evacuation, given by the number of shelters that are being used for the evacuation. If the planners decide to use a certain possible shelter, we call this shelter opened.

2.2. Assumptions

In the following, we state the underlying assumptions for the model we consider.

(A1) *We consider individual car traffic and organized public bus transports.*

Most people will try to get to the safe places using their own car if possible. However, there are also people that do not have access to private transportation and rely on public transportation via buses. Examples for such evacuees are old people, people who do not own a car or tourists and commuters who came to the endangered area via public transportation. Therefore, we consider public transportation by bus and private transportation by cars simultaneously. In small-scale evacuations like evacuation of buildings evacuees might leave the endangered zone on foot as well. Since we focus on large-scale evacuation our model will not include pedestrian evacuation.

(A2) *Individual traffic appears in the endangered region and then drives to the open shelters. Shelters need to provide sufficient parking space and evacuee capacity.*

Every shelter that is used during the evacuation must have enough capacity to hold and supply all evacuees that try to reach this shelter no matter how they go there. Those arriving by car will leave their car in the vicinity of the shelter. Therefore, the shelter must provide enough parking space.

(A3) *All buses start at a depot, pick up evacuees at designated collection points, and bring them to shelters. A bus might continue picking up people and bringing them to shelters until everybody is on safe ground. No parking space is needed at the shelters, but there must be enough room left to accommodate the evacuees.*

Initially buses are in a depot where they are prepared for the evacuation and their drivers are instructed. Each bus can pick up people at several collection points before going to a shelter. Once a bus reaches a shelter all evacuees will disembark the bus. In an emergency situation it would not be possible to persuade people to stay in the bus and go to another shelter once they have reached a shelter which can accommodate them. In a large-scale evacuation the number of buses will most likely be too small to get all evacuees out of the danger zone with a single trip of each bus. Probably buses will have to return to the danger zone after dropping people at shelters and pickup more evacuees. The places where the buses will pickup people must be known prior to the evacuation so that people know where to go when they get the request to evacuate.

(A4) *Evacuees start their evacuation at different times.*

The people that have to be evacuated will not appear all at the same time. Some people appear later during the evacuation because families or other groups might gather first in order to evacuate together, people do not immediately realize that they have to be evacuated, or people first have to reach places from which they can be evacuated further.

(A5) *The planner has full control over both the buses and the individual traffic.*

Most evacuation problems follow either a user-optimal, or a system-optimal approach. While the former aims at finding solutions that model selfish behavior of participants (i.e., every agent takes the evacuation route that is best for himself), the latter aims at finding solutions that optimize the performance of a solution, as if the behavior of every participant could be controlled by the planning authority.

This paper presents a system-optimal approach. Its usage can be justified, as (a) the routes of the public transport are indeed determined by the evacuation planner; (b) individual traffic can be influenced in their route choice via security forces and static traffic guidance; and (c) optimization results can be complemented with simulation models within the Decision Support System that this model is part of, following the sandwich approach [HHK$^+$11].

(A6) *Planning aspects are the duration of the evacuation, the risk of the evacuees, and the number of shelters opened.*

The ultimate goal of an evacuation is in most cases the time it takes to evacuate people. However, depending on the type of disaster the risk of the used routes can also be important. If some of the buildings in the affected zone are damaged and might collapse, then routes along these buildings should only be used if taking other routes would significantly increase the evacuation time and, thus, expose evacuees to danger much longer. Due to this reasoning we consider the risk of the routes as a second objective of the evacuation process. A third aspect of the evacuation is the economic cost of the evacuation. This cost is strongly linked to the number of shelters that are used. The more shelters that are used, the more forces are needed to maintain and supply these shelters. Therefore the number of used shelters should be chosen as small as possible while not increasing risk or evacuation time to badly.

(A7) *Vehicles may cause congestion.*

The evacuation is an extreme situation for the street network. Since there are many more cars than usual, there will probably be congestion. The congestion depends on the size of the street and on the number of cars trying to use the street. Since we consider a system optimum approach, we model the congestion by capacitated roads (see for example [SCH91]).

2.3. Model and Parameters

We now describe an optimization model that includes the assumptions of the previous section. All notation we introduce here is summarized in Appendix A.

The streets that can be used during the evacuation are modelled using a time expanded network consisting of a directed graph $G = (\mathcal{N}, \mathcal{A})$ with travel times δ_{ij}^t, arc capacity κ_{ij}^t and risk value r_{ij}^t for each arc $(i,j) \in \mathcal{A}$ and time step t. The capacity of a street is mainly determined by its size. Collapsed buildings or other events might reduce the size of a street, but these events are not predictable. Therefore, we model them using the risk value. In this network, congestion occurs when evacuees try to access a street which has already reached its capacity. To this end, we use holdover arcs in every node with unlimited capacity to model vehicles waiting for another edge to have sufficient capacity again.

The considered time horizon is denoted by $\mathcal{T}$. This is not the evacuation time we are aiming for, but an upper bound on the evacuation time that is needed by our model. This quantity is used to build the time expanded network.

For public transportation we assume that there is already an established set of collection points $\mathcal{N}^B \subseteq \mathcal{N}$ where evacuees gather for further transportation to shelters. For each collection point it is known how many people will appear at this point in each time step. This quantity is denoted by l_i^{Bt} for collection points $i \in \mathcal{N}^B$ and time step $t \in \mathcal{T}$. In contrast to the model used in [GG14, GGH13] we do not require the number of people waiting at each collection point to be given in multiples of bus loads. For private transportation we assume the same, i.e. it is known how many people will start the evacuation by car in each demand point from a set $\mathcal{N}^I \subseteq \mathcal{N}$ in each time step. This number of people is given by l_i^{It} for node $i \in \mathcal{N}^I$ and time step $t \in \mathcal{T}$. Again this needs not be given in car loads.

We are given a set of possible shelter location $\mathcal{N}^S \subseteq \mathcal{N}$. For each such location $i \in \mathcal{N}^S$ we are given the number of people u_i that this shelter can hold and additionally the parking space u_i^I available near this shelter.

The set of buses available for the public evacuation transit is denoted by $\mathcal{B}$. For simplicity, we assume that all buses have the same capacity C (however, different capacities can easily be included in our model). All buses start at a depot node $d \in \mathcal{N}$. Since a bus uses more space on a street than a car, we also have a parameter α which denotes the ratio of edge capacity occupied by a person travelling by car compared to the edge capacity occupied by a person travelling by bus. We assume that this ratio is the same for all people, i.e. all cars carry the same number of people.

Once the used shelter locations have been chosen, the public and private traffic simultaneously route to these shelters. The private traffic is modelled as a dynamic network flow, the public traffic (the buses) as a dynamic multicommodity network flow. The private traffic is a single commodity whereas each bus is a commodity of its own. The flow of the buses has to be chosen such that all people that need public transportation can be brought to shelter locations while respecting the bus capacity. Both flows are chosen simultaneously in a system optimal way.

The total risk exposure is given by the sum of the risks of the individual arcs over all time steps. The risk of a single arc at a time step is given by the risk value of the arc multiplied with the number of people on this arc at this time step.

Formulating these aspects mathematically, we propose the following multi-criteria mixed-integer programming model, which we call the Comprehensive Evacuation Problem (CEP).

$$\min\ (\Delta, R, S) \tag{1}$$

$$\text{s.t. } \Delta \geq t z^t \qquad \forall t \in \mathcal{T} \tag{2}$$

$$\Delta \geq \sum_{(i,j)\in\mathcal{A}} \sum_{t\in\mathcal{T}} \delta_{ij}^t x_{ij}^{bt} \qquad \forall b \in \mathcal{B} \tag{3}$$

$$R \geq \sum_{t\in\mathcal{T}} \sum_{(i,j)\in\mathcal{A}} r_{ij}^t \left(f_{ij}^t + \sum_{b\in\mathcal{B}} g_{ij}^{bt} \right) \tag{4}$$

$$\sum_{i\in\mathcal{N}} y_i \leq S \tag{5}$$

$$\sum_{i:(i,j)\in\mathcal{A}} x_{ij}^{bt} \geq \sum_{k:(j,k)\in\mathcal{A}} x_{jk}^{b(t+\delta_{jk}^t)} \qquad \forall b \in \mathcal{B}, t \in \mathcal{T}, j \in \mathcal{N} \tag{6}$$

$$\sum_{i:(d,i)\in\mathcal{A}} \sum_{t\in\mathcal{T}} x_{di}^{bt} \leq 1 \qquad \forall b \in \mathcal{B} \tag{7}$$

$$v_j^{It} - w_j^{It} + \sum_{i:(i,j)\in\mathcal{A}} f_{ij}^t = \sum_{k:(j,k)\in\mathcal{A}} f_{ij}^{t+\delta_{jk}^t} \qquad \forall t \in \mathcal{T}, j \in \mathcal{N} \tag{8}$$

$$\alpha f_{ij}^t \leq \kappa_{ij}^t z^t \qquad \forall t \in \mathcal{T}, (i,j) \in \mathcal{A} \tag{9}$$

$$\sum_{s=1}^{t} v_j^{Is} \leq \sum_{s=1}^{t} l_j^{Is} \qquad \forall t \in \mathcal{T}, j \in \mathcal{N}^I \tag{10}$$

$$\sum_{t\in\mathcal{T}} v_j^{It} = \sum_{t\in\mathcal{T}} l_j^{Is} \qquad \forall j \in \mathcal{N}^I \tag{11}$$

$$v_j^{bt} - w_j^{bt} + \sum_{i:(i,j)\in\mathcal{A}} g_{ij}^{bt} = \sum_{k:(j,k)\in\mathcal{A}} g_{ij}^{b(t+\delta_{jk}^t)} \qquad \forall b \in \mathcal{B}, t \in \mathcal{T}, j \in \mathcal{N} \tag{12}$$

$$g_{ij}^{bt} \leq C x_{ij}^{bt} \qquad \forall b \in \mathcal{B}, t \in \mathcal{T}, (i,j) \in \mathcal{A} \tag{13}$$

$$\sum_{s=1}^{t} \sum_{b\in\mathcal{B}} v_j^{bs} \leq \sum_{s=1}^{t} l_j^{Bs} \qquad \forall t \in \mathcal{T}, j \in \mathcal{N}^B \tag{14}$$

$$\sum_{t\in\mathcal{T}} \sum_{b\in\mathcal{B}} v_j^{bt} = \sum_{t\in\mathcal{T}} l_j^{Bs} \qquad \forall j \in \mathcal{N}^B \tag{15}$$

$$\alpha f_{ij}^t + \sum_{b\in\mathcal{B}} x_{ij}^{bt} \leq \kappa_{ij}^t \qquad \forall t \in \mathcal{T}(i,j) \in \mathcal{A} \tag{16}$$

$$\sum_{t\in\mathcal{T}} w_j^{It} \leq u_j^I y_j \qquad \forall j \in \mathcal{N}^S \tag{17}$$

$$\sum_{t\in\mathcal{T}} w_j^{It} + \sum_{b\in\mathcal{B}} w_j^{bt} \leq u_j y_j \qquad \forall j \in \mathcal{N}^S \tag{18}$$

$$v_i^{bt} = 0 \qquad \forall b \in \mathcal{B}, t \in \mathcal{T}, i \in \mathcal{N} \setminus \mathcal{N}^B \tag{19}$$

$$v_i^{It} = 0 \qquad \forall t \in \mathcal{T}, i \in \mathcal{N} \setminus \mathcal{N}^I \tag{20}$$

$$w_i^{bt} = 0 \qquad \forall b \in \mathcal{B}, t \in \mathcal{T}, i \in \mathcal{N} \setminus \mathcal{N}^S \tag{21}$$

$$w_i^{It} = 0 \qquad \forall t \in \mathcal{T}, i \in \mathcal{N} \setminus \mathcal{N}^S \tag{22}$$

$$f_{ij}^t \geq 0 \qquad \forall t \in \mathcal{T}, (i,j) \in \mathcal{A} \tag{23}$$

$$g_{ij}^{bt} \geq 0 \qquad \forall b \in \mathcal{B}, t \in \mathcal{T}, (i,j) \in \mathcal{A} \tag{24}$$

$$x_{ij}^{bt} \in \mathbb{B} \qquad \forall b \in \mathcal{B}, t \in \mathcal{T}, (i,j) \in \mathcal{A} \tag{25}$$

$$z^t \in \mathbb{B} \qquad \forall t \in \mathcal{T} \tag{26}$$

$$v_i^{bt}, w_i^{bt} \geq 0 \qquad \forall b \in \mathcal{B}, t \in \mathcal{T}, i \in \mathcal{N} \tag{27}$$

$$v_i^{It}, w_i^{It} \geq 0 \qquad \forall t \in \mathcal{T}, i \in \mathcal{N} \tag{28}$$

$$y_i \in \mathbb{B} \qquad \forall i \in \mathcal{N}_S \tag{29}$$

$$\Delta, R, S \geq 0 \tag{30}$$

In this mixed integer program we use the following variables: f_{ij}^t denotes the number of evacuees using cars passing arc (i,j) at time t. In contrast, g_{ij}^{bt} denotes the number of evacuees using bus b to go from node i to node j at time t. Variable x_{ij}^{bt} is 1 if and only if bus b is going from i to j at time t. z^t is a variable which helps determining the time step at which the last evacuee travelling by car reaches his or her shelter. Variables v_i^{bt} (v_i^{It}) denote the number of evacuees that start their evacuation by bus b (by car) at time t in node i. Analogously variables w_i^{bt} and w_i^{It} denote the number of evacuees ending their evacuation in node i at time t. A shelter i is opened if and only if $y_i = 1$.

The objective (1) is to minimize the evacuation time Δ, the risk R and the number of used shelters S. These objectives are computed using constraints (2) – (5). Constraints (2) and (3) ensure that Δ is the maximal evacuation time. The risk R depends on the number of people passing a link. This relation is expressed in constraint (4). For the bus routing we ensure that a bus can only stop once and does not reappear (6) and that each bus starts at the depot d (if it starts at all) (7). The number of used shelters is determined in constraint (5).

The individual traffic is modelled by constraints (8) – (11). Constraint (8) is the flow conservation constraint for the individual traffic flow and constraint (9) ensures that $z^t = 1$ for all t at which there is individual traffic in the network. The start of evacuees using cars is modelled in constraints (10) and (11). Constraint (10) ensures that the number of evacuees that have already departed until time t is not larger than the number of evacuees that have occurred until time t and constrain (11) ensures that in the end all evacuees appearing at node i have in fact started their evacuation.

Similarly to the individual traffic flow we also have a passenger flow for each bus. This flow is modelled in constraints (12) – (15). The flow conservation constraint for this flow for each bus is given by constraint (12). Constraint (13) restricts the passenger flow to arcs that the bus actually uses. Similarly to the evacuees that are travelling by car, the evacuees travelling by bus have to start their evacuation at some point in time. That is why the structure of constraints (14) and (15) resemble the structure of constraints (10) and (11).

The individual and the public traffic are linked together in the edge capacity constraints (16). Each used shelter must supply enough parking space and enough room to support evacuees. These two aspects are modelled in constraints (17) and (18). Finally, evacuees may only be picked up at nodes where they appear and may only be dropped off at shelter nodes. This is modelled in constraints (19) – (22).

Note that CEP is NP-hard (containing the bus evacuation problem [GGH13, GG14] as a special case).

3. A Genetic Algorithm for the CEP

As the proposed mixed-integer program is not likely to be solvable in sufficient time for instances of real-world size, we propose a heuristic solution procedure in the following. Genetic algorithms have been established as a method of choice for large-scale multi-criteria problems due to the possibility of calculating a set of solutions in one run.

3.1. Solution Representation

We begin with describing how solutions to the CEP – of which some variables may have continuous values – can be modeled using a discrete genotype. The main idea is to precalculate possibly interesting sets of paths, and to assign these paths to discrete flow units.

As before, we denote by $\mathcal{N}^I \subseteq \mathcal{N}$ nodes of the network where individual traffic appears, and by $\mathcal{N}^B \subseteq \mathcal{N}$ nodes where transit-dependent evacuees appear. For each (i,j) in $(\mathcal{N}^I \cup \mathcal{N}^B) \times \mathcal{N}^S$, and in $(\mathcal{N}^S \cup \{d\}) \times \mathcal{N}^B$, determine a set of candidate paths $\mathcal{P}_{ij}$.

While the former path sets represent traffic from source nodes to sinks, the latter sets model bus routes from shelters back to collection points, and the first trip from the depot to collection points.

The more candidate paths are used, the more degrees of freedom does the genetic algorithm have. We determine the Pareto solutions with respect to time and risk by solving a series of resource-constrained shortest path problems in $(\mathcal{N}, \mathcal{A})$.

Using these paths $\mathcal{P}_{ij}$, we represent a solution to the CEP by discrete "flow packets". Again, the finer the granularity of the continuous flows, the more degrees of freedom does the genetic algorithm have.

For individual traffic, a flow packet F^I is given by (i, t, s), where $i \in \mathcal{N}^I$ is its source node, t is its earliest possible starting time, and s denotes its size (i.e., the amount of flow represented by this packet). We denote the set of flow packets by $\mathcal{F}^I$ and generate it by using all individual traffic that appears at a given node at a given time as a flow packet (e.g., when 3 units of flow appear at node i at time 0, and 2 units appear at node i at time 1, we model them using two flow packets $(i, 0, 3)$ and $(i, 1, 2)$). A solution for the individual flow is then given as the assignment of a shelter node $j \in \mathcal{N}^S$ and a corresponding path $\mathcal{P}_{ij}$ for all flow packets $(i, t, s) \in \mathcal{F}^I$.

For the bus transport, the solution representation is slightly more elaborate, as a single bus may perform several subsequent tours. As for the individual traffic, we collect transit-dependent evacuees in flow packets $F^B = (i, t, s)$. To enhance the capacity usage of buses, bus flow packets are not generated by using every appearance as a single packet, but evacuees are gathered instead, until an amount equal to the bus capacity have appeared. A solution to the bus transport problem aspect is then given by a sequence of bus packets for every bus, along with destination shelter nodes and a path for each bus packet from $\mathcal{F}^B$.

Note that a solution represented this way is feasible if and only if the resulting shelter usage is feasible. All dynamic flow aspects are neglected so far; i.e., to generate a full solution to the CEP (the phenotype) using such a solution representation, we need to determine at what times the flow uses each arc. Arc capacities may prevent packets to use their assigned path unhindered.

To this end, we employ the following strategy to generate a phenotype from the above representation of a solution. For each package, we calculate the expected arrival time under the assumption that the routing is not hindered by any capacity conflicts. Bus packets $\mathcal{F}^B$ and individual flow packets $\mathcal{F}^I$ are then sorted according to their arrival time in descending order. Then, first the bus packets, then the individual flow packets are routed in a greedy fashion, i.e., packets with late expected arrival time are routed first, and when a desired arc has not sufficient remaining capacity at a given point in time, the respective packet waits at its current node until the required amount of capacity becomes available again.

3.2. Algorithm Mechanisms

A wide variety of genetic algorithms have been proposed for multi-objective optimization problems; see, e.g, the survey [KCS06]. In the following, we outline the basic mechanisms of our proposed algorithm that follow the guidelines of the well-known NSGA-II method [DPAM02].

Starting Population. A feasible starting population is generated the following way: The minimal number of shelters n_S needed to accommodate all evacuees is determined with the help of a small integer program (see Appendix B). We generate four different shelter assignments using the integer program described in Appendix C: An assignment that minimizes the risk when using n_S shelters, one that minimizes the duration when using n_S shelters, and the same for $n_S + 1$ shelters. For each of these four assignments, paths are chosen randomly for each packet from the precalculated path sets.

The reasoning to produce starting solutions with small numbers of used shelters is that it is "easy" for the genetic algorithm to use an additional shelter, but "hard" to close a shelter (as the former operation only requires one packet to be routed to the shelter, while the latter operation needs all packets routed to the shelter to be assigned someplace else).

In iteration τ, we denote the current population of solutions by P_τ with cardinality N. Using mutation and crossover, we determine a new population P'_τ consisting of $2N$ candidate solutions. From these $2N$ solutions, we form the new generation $P_{\tau+1}$ using the following mechanisms:

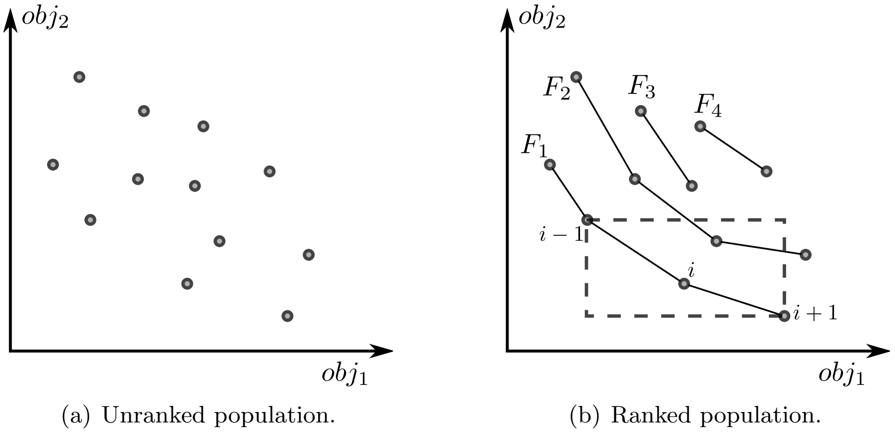
Fitness assignment. Given a population P'_τ of solutions, we need to determine a fitness value for each solution to determine its likelihood to procreate into the next generation. We follow the approach to assign a rank based on non-dominated sorting; i.e., in step $k = 1$ we determine all non-dominated solutions with respect to the objectives risk, evacuation time, and number of opened shelters. These solutions are noted as F_1. In step k, while $R_k := P'_\tau \setminus \cup_{i=1,\dots,k} F_i \neq \emptyset$, we determine the set F_{k+1} as all non-dominated solutions in R_k.

Diversity mechanism. After the non-dominated sorting, we determine a crowding value for each solution in P'_τ as the relative distance in each objective dimension to its neighbors from the respective non-dominated front F_i. To create $P_{\tau+1}$, solutions are chosen using a series of 2-tournaments. The winner in each comparison is lexicographically determined by first comparing the fitness rank, then comparing the crowding distance.

Note that elitist solutions are not stored externally, but non-dominated solutions are always given priority over any dominated solutions. Thus, elites are kept within the population.

The fitness assignment and calculation of the crowding distance are exemplified in Figure 1 (more details can be found in [KCS06]). We use only two criteria for simplicity.

In Figure 1(a), the objective values of the current population are shown. The corresponding ranking in $F_1, \ldots, F_4$ is given in Figure 1(b). The dashed box indicates the calculation of the crowding distance for solution i, which uses the two neighbors $i-1$ and $i+1$ from the same front F_1.



(a) Unranked population.

(b) Ranked population.

Figure 1: Ranking method for the genetic algorithm.

Mutation and Crossovers. Given two solutions, we generate two new solutions by exchanging the individual traffic and the bus traffic parts of the solutions, respectively. Then, these solutions are randomly mutated using the following neighborhoods: (a) For each car packet, a different shelter and a different path may be assigned. (b) For each bus packet, a different shelter and a different path may be assigned. If a different shelter is assigned and the bus serving this packet has subsequent trips, then also the path to the next bus packet needs to be changed. (c) A bus packet can be assigned to any bus at any position within the current schedule. As before, further path adjustments may become necessary.

Local Search. Finally, solutions are post-optimized using a local search based on the same neighborhood as used for the random mutations. To reduced computation times, local search is performed until a local optimum is reached, or a given number of iterations bas been performed.

3.3. Algorithm Summary

Algorithm 1 recapitulates the main aspects of the proposed genetic algorithm.

4. Data Aggregation

Even though the proposed genetic algorithm is able to handle large-scale evacuation problems, results may be further improved by considering instances of reduced size

Algorithm 1 Genetic Algorithm for CEP

```
Require: An instance of CEP.
Preprocessing:
      Determine flow packets $\mathcal{F}^I$, $\mathcal{F}^B$.
      Determine path sets $\mathcal{P}_{ij}$.
      Generate starting population.
while (computation time remaining) do
    Do non-dominated sorting.
    Calculate crowding distances.
    Fill in next generation.
    Crossover and mutate.
    Local search.
end while
return Set of solutions.
```

instead.

This touches the fundamental question to determine *what the "real" instance would be.* This aspect is only implicitly included in many research on evacuation planning, insofar that a single instance is presented which is supposed to be "the" real-world instance. We take a fundamentally different approach by considering instances modelling a real-world evacuation with different degree of detail. A solution that is produced for an aggregated instance is then extended to a solution to the most-detailed (and thus most-realistic) available graph.

4.1. Graph Reduction

We first describe how a given CEP instance can be aggregated to a reduced CEP instance.

Given a graph with travel time and risk for every arc, we create supernodes consisting of several subnodes and edges. Figure 2 shows an example for such a reduction. Here, nodes i_2, i_3 and i_4 are reduced to a single supernode that also contains the edges e_2, e_3, and e_4.

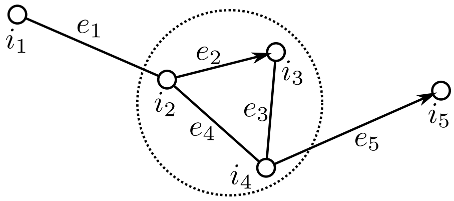


Figure 2: Graph reduction example.

Note that in this example, the supernode consists of a strongly connected graph. This is a necessary requirement in the following since we need to be able to compute paths inside these supernodes.

These graph reductions are achieved by assigning an "importance" value v_e to each edge e, that represents the estimated impact such an edge has when solving the instance. We find these values by setting

$$v_e = \frac{r_e - r_e^{\min}}{r_e^{\max} - r_e^{\min}} + \frac{\delta_e - \delta_e^{\min}}{\delta_e^{\max} - \delta_e^{\min}},$$

where $r_e^{\max} = \max_{e\in\mathcal{A}} r_e$, $r_e^{\min} = \min_{e\in\mathcal{A}} r_e$, and analogously for $\delta^{\max}$ and $\delta^{\min}$, respectively. Thus, the higher the value v_e for an edge, the more is it expected to have an influence on the quality of a solution.

We sort all edges with respect to v_e, and eliminate them in ascending order until a desired number of nodes (or edges) is reached. An edge is eliminated by subsuming both end nodes to the same supernode. There are two cases in which an edge must not be eliminated: If both end points are in $\mathcal{N}^I \cup \mathcal{N}^B \cup \mathcal{N}^S$ or if the resulting supernode is not strongly connected, i.e. the edge represents a one way street. If an edge cannot be eliminated we continue with the next edge. However, it can happen that such an edge may be eliminated at a later point.

Whenever a source node becomes part of a supernode, the demand is transferred to the supernode. The same happens for shelter locations: If a supernode contains a shelter, the supernode itself is used as a shelter.

4.2. Solution Extension

After solving an instance aggregated in the above way, we need to extend each solution to a solution for the original instance. Therefore, we have to extend the path from the reduced network to the original network. Consider Figure 3 as a guideline. A path from supernode sn_1 to sn_4 along the edges e_1, e_2, and e_3 needs to be extended to a path from s to t.

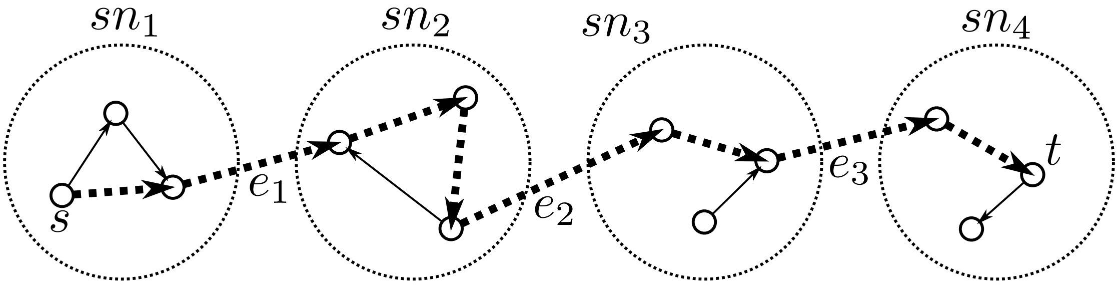


Figure 3: Extending an aggregated solution.

A path needs to be determined from s to the start of edge e_1 within sn_1, then from the end of e_1 to the start of e_2 within sn_2. In each supernode the shortest path is used. Finally, within the final supernode, another subpath needs to be determined from the end of edge e_3 to t. This way, every path used in the aggregated instance is extended to a path in the original instance.

5. Computational Results

We present experimental results on two real-world instances: The first instance describes the evacuation of a city region due to a bomb defusion. Here, the evacuation region is typically circular with the bomb in the center, and risk is determined by distance to the evacuation center. The city of Kaiserslautern, Germany, is used as an example. The second instance describes the evacuation of a coastal city due to an earthquake with a subsequent flood. Here, the city of Nice, France, is used for our experiments.

All experiments were conducted on a computer with a 16-core Intel Xeon E5-2670 processor, running at 2.60 GHz with 20MB cache, 32 GB RAM and Ubuntu 12.04. To solve (mixed) integer programs, we used the non-commercial solver SCIP version 3.1.0 with SoPlex 1.7.2 [Ach09]. The genetic algorithm was implemented with gcc v. 4.5.4. using the compile flag -O3. We use a population size of 1250 solutions.

For a comparison with the genetic algorithm proposed here, we also used CPLEX to solve the mixed-integer programm for the CEP directly. However, for instances with a size of 50 nodes upwards, CPLEX failed to produce feasible solutions. This is far below the realistic instance size we consider here, and thus only applicable to academic instances.

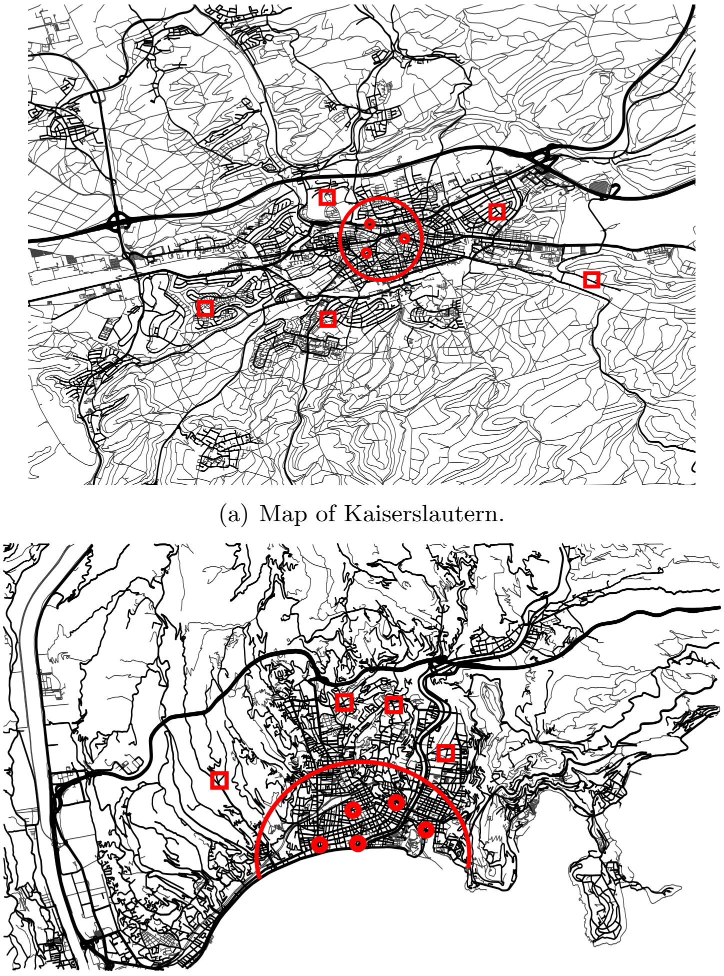
5.1. Instance 1: Kaiserslautern

The street network of Kaiserslautern in presented in Figure 4(a), based on OpenStreetMap (OSM) data. After removing irrelevant information for the purpose of our model (like road shapes), the corresponding road graph consists of 13284 nodes and 32463 arcs.

The scenario we assume is a bomb being found within the city center, as frequently happens at construction sites in German cities. The bomb defusion triggers an evacuation radius, which is usually circular with the bomb in the center. We identified five gymnasiums in the surrounding area as possible shelters, and calculated their capacity based on their area. We consider three pickup points for transit-dependent evacuees, and generate additional random car demand within the endangered region. In total, we consider 800 persons travelling by car, and 360 persons being evacuated by bus.

Using this setting, we consider three instances with different degree of aggregation: The original instance with 13284 nodes, a reduced instance with 8000 nodes, and a further reduced instance with only 4000 nodes. For each instance, we allow 30 minutes for the genetic algorithm, excluding pre- and post-processing time. We present both the estimated solution quality (as optimized by the algorithm), and the solution quality of the extended solution in the original network in Figure 5.

The total observed computation time including pre- and postprocessing is $2280s$ for 4000 nodes, $3525s$ for 8000 nodes, and $7618s$ for the full instance with 13284 nodes. A comparison of the populations shows that the solutions computed for the smallest instance already have a comparable quality to those of the full instance, with significantly less computational effort – in fact, solutions computed for the original instance may even



(a) Map of Kaiserslautern.

(b) Map of Nice.

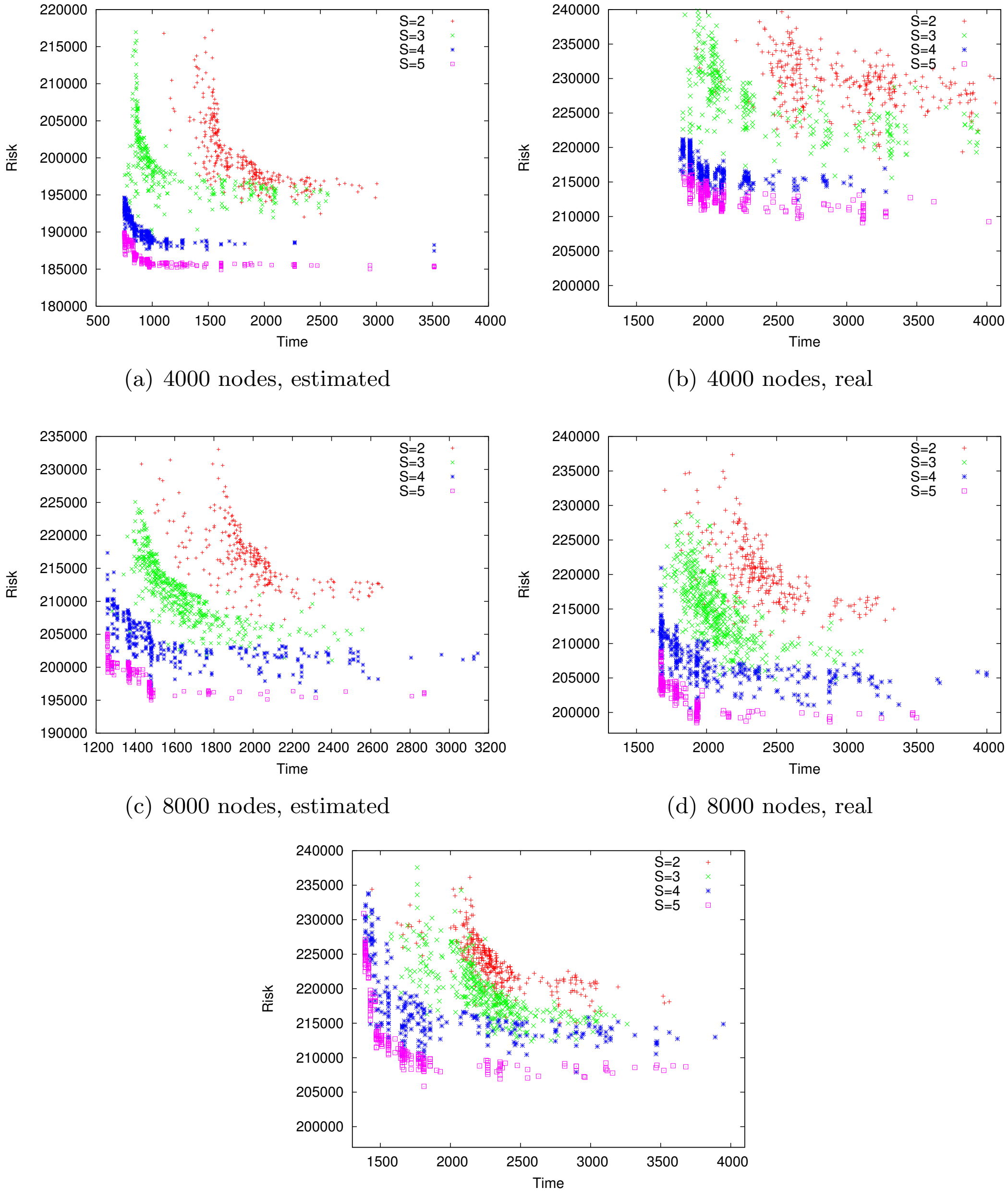
Figure 4: Evacuation instances.

have worse quality, due to the increased computational effort per generation during the 30 minutes reserved for the genetic algorithm. We find a trade-off: While finer granulated instances can potentially create better solutions, they also lead to less iterations in the genetic algorithm, and thus also worsen the achieved solution quality.

5.2. Instance 2: Nice

The second scenario we consider is an earthquake with a subsequent flood in the area of Nice, France. The OSM data related to the region is presented in Figure 4(b). The corresponding graph contains 6237 nodes and 13209 arcs. Risk is calculated using simulation methods to determine the likelihood of a building collapse[1].

[1] We thank the BRGM France for providing us with valuable risk data.



(a) 4000 nodes, estimated

(b) 4000 nodes, real

(c) 8000 nodes, estimated

(d) 8000 nodes, real

(e) 13284 nodes, real

Figure 5: Estimated and real objective values for Kaiserslautern.

We consider five collection points for buses, and distribute individual traffic sources along the endangered coastal region. Four possible shelters are identified within higher elevated areas. A total of 700 individually travelling evacuees and 300 transit-dependent evacuees are considered. We generated two further aggregated instances, one with 2000 nodes, and one with 4000 nodes. The results are presented in Figure 6.

We observe that for this instance, a reduction to 2000 nodes was too high to produce meaningful results. Specifically, the estimated and the real objective values have a high discrepancy (note that the scale of Figure 6(b) is different to the scale of Figures 6(d) and 6(e)).

For 4000 nodes on the other hand, we can observe even better results with respect to risk than when using the full instance with 6237 nodes.

The total observed computation time including pre- and postprocessing is $2020s$ for 2000 nodes, $4030s$ for 4000 nodes, and $6791s$ for the full instance with 6237 nodes.

6. Conclusion

We presented a comprehensive optimization model for evacuation planning, including both individual and public transport, as well as shelter location decisions. Not only evacuation time was taken into account, but also the risk for the evacuees. These aspects were modeled as a multi-criteria mixed-integer program.

Due to the computational complexity of such an integrated problem approach, we introduced a genetic algorithm to solve instances of realistic size heuristically. Furthermore, we explicitly discussed the impact of data aggregation on the solution quality, reflecting the trade-off between a finer granulation and smaller computation times.

Using real-world instances modelling Nice and Kaiserslautern, these approaches were evaluated.

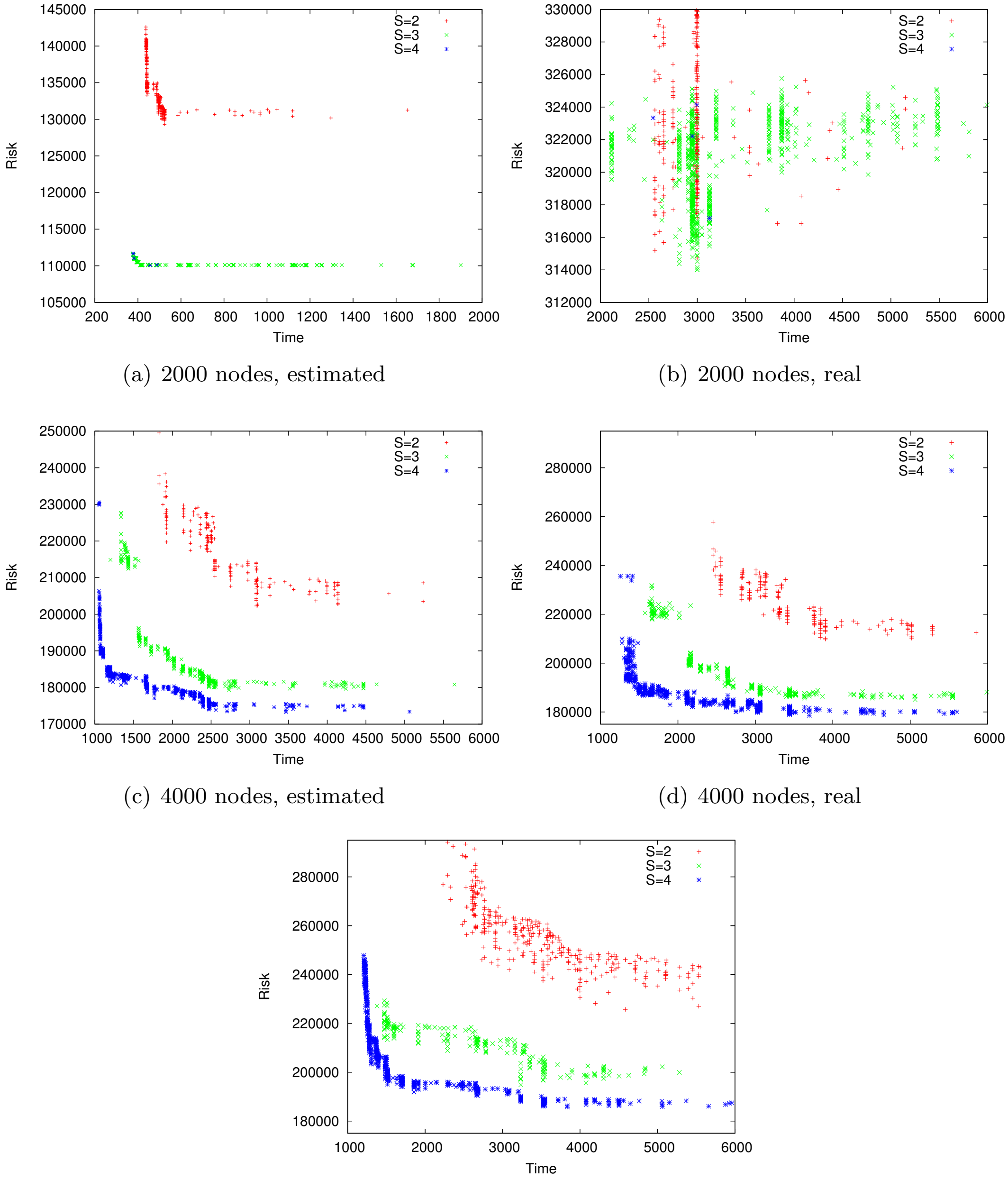
From a mathematical perspective, further research includes the comparison of different routing strategies, path sets, and sequential heuristic methods. From a practical perspective, solutions need to be evaluated considering further criteria like evacuation costs, personnel demand, and logistic effort to provide supplies. This is performed in subsequent steps within the Decision Support System.

References

[AAW10] H. Abdelgawad, B. Abdulhai, and M. Wahba, *Multiobjective optimization for multimodal evacuation*, Transportation Research Record: Journal of the Transportation Research Board **2196** (2010), no. 1, 21–33.

[Ach09] T. Achterberg, *SCIP: Solving constraint integer programs*, Mathematical Programming Computation **1** (2009), no. 1, 1–41.

[AI06] N. Altay and W. G. Green III, *OR/MS research in disaster operations management*, European Journal of Operational Research **175** (2006), no. 1, 475 – 493.



(a) 2000 nodes, estimated

(b) 2000 nodes, real

(c) 4000 nodes, estimated

(d) 4000 nodes, real

(e) 6237 nodes, real

Figure 6: Estimated and real objective values for Nice.

[Bis11] D. R. Bish, *Planning for a bus-based evacuation*, OR Spectrum **33** (2011), no. 3, 629–654.

[Bre12] S. Bretschneider, *Mathematical models for evacuation planning in urban areas*, Springer- Heidelberg New York Dordrecht London, 2012.

[CFS82] L.G. Chalmet, R.L. Francis, and P.B. Saunders, *Network models for building evacuation*, Fire Technology **18** (1982), no. 1, 90–113.

[CVVL02] C. A. Coello, D. A. Van Veldhuizen, and G. B. Lamont, *Evolutionary algorithms for solving multi-objective problems*, vol. 242, Springer, 2002.

[DGVW11] K. F. Doerner, W. J. Gutjahr, and L. Van Wassenhove, *Special issue on optimization in disaster relief*, OR Spectrum **33** (2011), no. 3, 445–449 (English).

[DPAM02] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, *A fast and elitist multiobjective genetic algorithm: NSGA-II*, Evolutionary Computation, IEEE Transactions on **6** (2002), no. 2, 182–197.

[FLRT08] R. L. Francis, T. J. Lowe, M. B. Rayco, and A. Tamir, *Aggregation Error for Location Models: Survey and Analysis*, Annals of Operations Research **167** (2008), no. 1, 171–208.

[GDT13] M. Goerigk, K. Deghdak, and V. T'Kindt, *A two-stage robustness approach to evacuation planning with buses*, Tech. report, Technische Universität Kaiserslautern, 2013, submitted.

[GG14] M. Goerigk and B. Grün, *A robust bus evacuation model with delayed scenario information*, OR Spectrum (2014), 1–26, doi:10.1007/s00291-014-0365-8.

[GGH13] M. Goerigk, B. Grün, and P. Heßler, *Branch and bound algorithms for the bus evacuation problem*, Computers & Operations Research **40** (2013), no. 12, 3010–3020.

[GPKM10] P. S. Georgiadou, I. A. Papazoglou, C. T. Kiranoudis, and N. C. Markatos, *Multi-objective evolutionary emergency response optimization for major accidents*, Journal of Hazardous Materials **178** (2010), no. 1–3, 792 – 803.

[HHK$^+$11] H.W. Hamacher, S. Heller, W. Klein, G. Köster, and S. Ruzika, *A sandwich approach for evacuation time bounds*, Pedestrian and Evacuation Dynamics (R. D. Peacock, E. D. Kuligowski, and J. D. Averill, eds.), Springer US, 2011, pp. 503–513.

[HK78] D. W. Hearn and H. W. Kuhn, *Network aggregation in transportation planning: A fixed point method for treating traffic equilibria*, vol. 2, US Transportation Systems Center, 1978.

[HT01] H. W. Hamacher and S. A. Tjandra, *Mathematical modeling of evacuation problems: A state of the art*, In Pedestrian and Evacuation Dynamics (Schreckinberg, M. and Sharma, S. D. eds) **1964** (2001), 227–266.

[KCS06] A. Konak, D. W. Coit, and A. E. Smith, *Multi-objective optimization using genetic algorithms: A tutorial*, Reliability Engineering & System Safety **91** (2006), no. 9, 992–1007.

[KCY05] S. Kongsomsaksakul, A. Chen, and C. Yang, *Shelter location-allocation model for flood evacuation planning*, Journal of the Eastern Asia Society for Transportation Studies **6** (2005), 4237–4252.

[LGS05] Q. Lu, B. George, and S. Shekhar, *Capacity constrained routing algorithms for evacuation planning: A summary of results*, Advances in Spatial and Temporal Databases (Claudia Bauzer Medeiros, MaxJ. Egenhofer, and Elisa Bertino, eds.), Lecture Notes in Computer Science, vol. 3633, Springer Berlin Heidelberg, 2005, pp. 291–307.

[LNXD12] A. C.Y. Li, L. Nozick, N. Xu, and R. Davidson, *Shelter location and transportation planning under hurricane conditions*, Transportation Research Part E: Logistics and Transportation Review **48** (2012), no. 4, 715 – 729.

[PP14] C. Prodhon and C. Prins, *A survey of recent research on location-routing problems*, European Journal of Operational Research (2014), available online first.

[RPWE91] D. F. Rogers, R. D. Plante, R. T. Wong, and J. R. Evans, *Aggregation and disaggregation techniques and methodology in optimization*, Operations Research (1991), 553–582.

[SCH91] H. D. Sherali, T. B. Carter, and A. G. Hobeika, *A location-allocation model and algorithm for evacuation planning under hurricane/flood conditions*, Transportation Research Part B: Methodological **25** (1991), no. 6, 439–452.

[She07] J.-B. Sheu, *Challenges of emergency logistics management*, Transportation Research Part E: Logistics and Transportation Review **43** (2007), no. 6, 655 – 659, Challenges of Emergency Logistics Management.

[SMT09] M. Saadatseresht, A. Mansourian, and M. Taleai, *Evacuation planning using multiobjective evolutionary optimization approach*, European Journal of Operational Research **198** (2009), no. 1, 305 – 314.

[TCH07] G.-H. Tzeng, H.-J. Cheng, and T. D. Huang, *Multi-objective optimal planning for designing relief delivery systems*, Transportation Research Part E: Logistics and Transportation Review **43** (2007), no. 6, 673 – 686.

[Yam96] T. Yamada, *A network flow approach to a city emergency evacuation planning*, International Journal of Systems Science **27** (1996), no. 10, 931–936.

[YAM08] M. Yusoff, J. Ariffin, and A. Mohamed, *Optimization approaches for macroscopic emergency evacuation planning: A survey*, Information Technology, ITSim, International Symposium, IEEE, 2008, pp. 1–7.

[YK07] W. Yi and A. Kumar, *Ant colony optimization for disaster relief operations*, Transportation Research Part E: Logistics and Transportation Review **43** (2007), no. 6, 660 – 672, Challenges of Emergency Logistics Management.

[Zip80] P. H. Zipkin, *Bounds for aggregating nodes in network problems*, Mathematical programming **19** (1980), 155–177.

[Zip82] ———, *Aggregation and disaggregation in convex network problems*, Networks **12** (1982), no. 2, 101–117.

A. List of Notation

Symbol	Meaning
$\mathcal{N}$	Network nodes
$\mathcal{A}$	Network arcs
$\mathcal{N}^B \subseteq \mathcal{N}$	Nodes where demand for public transit arrives (collection points)
$\mathcal{N}^I \subseteq \mathcal{N}$	Nodes where individual traffic appears
$\mathcal{N}^S \subseteq \mathcal{N}$	Shelter nodes
$\mathcal{B}$	Set of buses
κ_{ij}^t	Capacity of arc $(i,j) \in \mathcal{A}$ at time t
$\mathcal{T}$	Time horizon
δ_{ij}^t	Traversal time of arc $((i,j) \in \mathcal{A}$ at time t
r_{ij}^t	Traversal risk of arc $((i,j) \in \mathcal{A}$ at time t
C	Bus capacity
u_j^I	Parking space capacity of shelter $j \in \mathcal{N}^S$
u_j	Bed capacity of shelter $j \in \mathcal{N}^S$
α	Ratio of required bus/car edge capacity
l_j^{It}	Individual traffic demand arriving in node $j \in \mathcal{N}^I$ at time t
l_j^{Bt}	Public transit demand arriving in node $j \in \mathcal{N}^B$ at time t

B. Determining the minimal number of shelters

To generate starting solutions to the genetic algorithm with a minimal number of shelters, the following integer program is employed.

$$\begin{aligned}
\min\ & \sum_{j \in \mathcal{N}^S} x_j \\
\text{s.t.}\ & \sum_{j \in \mathcal{N}^S} y_{pj}^I = 1 && \forall p \in \mathcal{F}^I \\
& \sum_{j \in \mathcal{N}^S} y_{pj}^B = 1 && \forall p \in \mathcal{F}^B \\
& \sum_{p=(i,t,s) \in \mathcal{F}^I} s y_{pj}^I \leq u_j^I x_j && \forall j \in \mathcal{N}^S \\
& \sum_{p=(i,t,s) \in \mathcal{F}^I} s y_{pj}^I + \sum_{p=(i,t,s) \in \mathcal{F}^B} s y_{pj}^B \leq u_j x_j && \forall j \in \mathcal{N}^S \\
& x, y^B, y^I \in \mathbb{B}
\end{aligned}$$

The optimal objective value represents the smallest number of shelters that need to be opened to accommodate all evacuees.

C. Determining shelter assignments minimizing a distance value

To generate starting solutions to the genetic algorithm, the following integer program is employed.

$$
\begin{aligned}
\min \quad & \sum_{j \in \mathcal{N}^S} \left(\sum_{p \in \mathcal{F}^I} val_{pj} y^I_{pj} + \sum_{p \in \mathcal{F}^B} val_{pj} y^B_{pj} \right) \\
\text{s.t.} \quad & \sum_{j \in \mathcal{N}^S} y^I_{pj} = 1 && \forall p \in \mathcal{F}^I \\
& \sum_{j \in \mathcal{N}^S} y^B_{pj} = 1 && \forall p \in \mathcal{F}^B \\
& \sum_{p=(i,t,s) \in \mathcal{F}^I} s y^I_{pj} \leq u^I_j x_j && \forall j \in \mathcal{N}^S \\
& \sum_{p=(i,t,s) \in \mathcal{F}^I} s y^I_{pj} + \sum_{p=(i,t,s) \in \mathcal{F}^B} s y^B_{pj} \leq u_j x_j && \forall j \in \mathcal{N}^S \\
& \sum_{j \in \mathcal{N}^S} x_j \leq n_S \\
& x, y^B, y^I \in \mathbb{B}
\end{aligned}
$$

Here, n_S is a given value that determines the maximum number of shelters that may be used. val_{pj} may be the distance of the shortest path connecting package p with shelter j, or the risk of the least risky such path.