Towards Evacuation Planning of Groups with Genetic Algorithms

Bjørnar Hansen Leonard Loland Morten Goodwin
Ole-Christoffer Granmo
Department of ICT, University of Agder,
Grimstad, Norway

Abstract

In crisis situations on board ships, it is of utmost importance to have the passengers safely evacuate to the lifeboats in an efficient manner. Existing methods such as marked escape routes and maps are not optimal as pre-planned escape routes may become heavily congested by passengers. Further, the closest lifeboat is not always feasible as lifeboat capacity can be exceeded. Also considering that some evacuees are strongly affiliated, such as families, and that they prefer to evacuate together as a group, it becomes a difficult problem to solve.

This paper models the area to be evacuated as a time-expanded graph with hazard severities as probabilities of survivability for each node. The presented approach applies a multi-objective genetic algorithm with multiple fitness functions to maximize the over all survivability. Finally, the proposed method picks the best evacuation plan from a pool of potential solutions returned by the genetic algorithm.

The solution generates better routing plans than comparable methods, specially in situations where grouping and congestions are considered. In essence this is an essential step towards automatic planning of evacuations which in turn contributes to smoother evacuations of crises situations and saving lives.

1 Introduction

In an ongoing crisis situation, on ships and elsewhere, many challenges have to be faced during evacuation. In the case of fire, as it spreads over time, it produces an ever-increasing amount of lethal heat and smoke, rendering rooms and corridors hazardous or unusable for evacuation. Further, emergency response teams can be late and may not have the capacity to assist everyone efficiently. A direct consequence is that people are often initially left to themselves in evacuation situations. The closest emergency exits may become heavily congested as masses of people converge on them, while the nearest lifeboats quickly reach their maximum capacity, forcing other evacuees to make detours to search for alternatives. It is also possible that escape routes are rendered too dangerous or unusable, and alternate, perhaps non-obvious, routes have to be used. On top of it all, the information required to make the best course of action, such as the locations of people and hazards, may not be available, or be erroneous, during the crisis. All this may lead to valuable time and resources being wasted.

While traditional static signs are meant to guide evacuees safely towards exits, they have shortcomings. They do not change if the evacuation routes become blocked or hazardous. In addition, if too many evacuees decide to take the same escape route, it leads to congestion and overcrowding.

To mitigate the problems that arise during crisis evacuation, research is being conducted on how personal electronic devices—such as smart phones—equipped with sensors can be applied for management of such situations [1, 12, 5]. Their built-in sensors and communication technologies can both gather information and share it among devices [8], and the aggregate of this information can contribute to clarify the current situation.

By leveraging this kind of real-time information, an automatic evacuation planning system can help resolve some of the challenges faced during a crisis situation, namely how to avoid casualties, congestion, and confusion. It can automatically determine escape routes for everyone present, taking care to lead evacuees away from hazardous situations, and avoid congestion by taking into account all passengers and their respective escape routes.

1.1 Problem Formulation

Fast, efficient and safe evacuation is important during crisis situations. Whereas current approaches to evacuation planning include pre-planned routes, a benefit could be had from providing real-time evacuation planning. Pre-evacuation planning is limited in that it cannot take into account the particularities of crisis situations as they happen; consequently, evacuation operation can be inefficient and unnecessarily dangerous.

Furthermore, it is of interest to take more of human behavior into account than has been done in related work. Specifically, group affiliation is an important aspect of human life, and it affects the evacuation process.

In line with common practice [6] we treat the escape as a time expanded directed graph of nodes and edges: $G(N, E)$. For this, any node $n_i \in N$

- is either a room (source), lifeboat (sink), super source or super sink,
- holds zero, one or more evacuees without exceeding its capacity $c(n)$, and
- has a survivability $\sigma(n)$ so that $\sigma(n) \in [0, 1]$ indicating the probability of survival for one time step.

Any potential flow from node n_i to node n_j is represented with an edge $e_{i,j} \in E$. A search space s represents a solution, i.e. paths consisting of edges and nodes, for every evacuee.

Additionally, an edge $e_{i,j}$ has capacity $c(e_{i.j})$ and flow value $f(e_{i,j})$. Further this paper extends the common terminology with node congestion $con(n_i)$. While the capacity $c(e_{i,j})$ reduces the flow of which a quantity moves from node n_i to n_j, $con(n_i)$ limits the number of people that can fit inside a room n_i. This is a realistic extension as there are practical limitations to how many people can be in a room at the same time.

The problem relies upon two main functions, namely the overall survivability of a search space, s:

$$f(s) = \sum_{n_i \in s} (\sigma(n_i)) \tag{1}$$

and the overall grouping of a search space, s:

$$g(s) = \sum_{e_{i,j} \in s} \tau(e_{i,j}, s) \tag{2}$$

where

$$\tau(e_{i,j}, s) = \begin{cases} 1 \ \textit{if } e_{i,j} \textit{ occours at least twice in } s \textit{ for the same group} \\ 0 \ \textit{otherwise} \end{cases} \tag{3}$$

The latter can be informally written as counting the number of edges overlapping within each group. An overlap occurs whenever at least two people use the same path at the same time.

Consequently, this paper has two objectives in prioritized order:

1. Maximizing the survivability: This is formalised, in line with common practice [5] as finding a search space $s^* \in \mathbf{S}$ so that $f(s^*) \geq f(s) \forall_s \in \mathbf{S}$. I.e. maximizing the probability that persons survive in the path chosen for them.
2. Maximizing the grouping: Finding a search space $s^* \in \mathbf{S}$ so that $g(s^*) \geq g(s) \forall_s \in \mathbf{S}$. I.e. maximizing the probability that groups stay together.

Note that we assume information such as physical layout of the ship, locations of people and lifeboats, affiliations and survivability in each room, is known. The authors realize that this may appear as an unrealistic assumption, but note significant effort is being made to collect and aggregate hazard information in similar scenarios from both stationary and smart phone sensors [12, 14].

1.2 Outline

Section 2 describes the related evacuation and evacuation planning. Section 3 continues with genetic algorithms specifically how it was used to solve the evacuation problem. Empirical results in a simulated ship environment are presented in section 4. Finally, conclusion and further work are presented in section 5.

2 Evacuation

Much work exists in the literature on evacuation modelling [11]. Common for evacuations is a five stage process [15]: (1) An alert is raised. (2) The persons present react to the alert. (3) A decision is made to evacuate. (4) The actual evacuation. (5) Verify that everyone has made it to safety.

Most existing work on evacuation planning focuses on off-line solutions aiding step (3) and (4). One of the main lines of research focus on mathematical modelling and solutions based on finding maximal flow in networks [4]. Other work has been based themselves on shortest path in a graph [9], while some recent research has been carried out for stochastic methods for planning safe escape routes [5].

2.1 Group Behavior in Crisis Situations

Groupwise evacuation is grounded in recent social theory. According to the "social attachment" model of human behavior during crisis situations [10], in threatening situations people tend to seek affiliation with familiar persons or attachment figures. This behavior delays the evacuation process; in fact, it has been shown to cause the loss of human lives because people linger together with their group or search for attachment figures instead of promptly evacuating. Evacuation planning without taking into account the strong force of group affiliation would be nigh on pointless, as it is unlikely that evacuees would follow a plan that required group members to go separate ways.

Furthermore, the social attachment model goes against earlier mass panic theories, which claim that chaotic human behavior is the norm when disaster strikes [3]. In contrast those earlier theories, the social attachment model describes evacuation as orderly in most cases. This certainly indicates a higher probability of evacuees displaying an ability to follow the dynamically planned routes than if they were panicking and behaving irrationally.

2.2 Evacuation Planning With Multi-objective Genetic Algorithms

Genetic algorithms (GA) have been used within the field of evacuation previously. In [13], the evacuation planning process is described as a three-part process which is performed as a preparatory measure for the case where actual evacuation is needed: Selecting safe areas, finding optimal paths from buildings to safe areas, and selecting the best safe area for each building is included in planning. The first step, selecting safe areas, was done manually. Next, the optimal paths from buildings to safe areas were determined according to safety and traffic. The last step, selecting the best evacuation routes for each building, was then solved with a genetic algorithm.

Kongsomsaksakul et al. [7] also consider predisaster evacuation planning. In their model, the problem is formulated as a Stackelberg game, where the leader is the evacuation planning authority designating shelter locations. The follower is the collection of evacuees, who according to the given shelter locations determine which shelter to move to and by which path.

The GA is employed by the planning authority to place shelters. Given a potential solution from the leader, the evacuees decisions are calculated. The result is fed into the GA's fitness function, which is a weighted sum of constraints on egress time, congestion, and shelter capacity.

3 Genetic Algorithm (GA)

The solution bases itself on an initial population of solutions that is further improved by iterations. Solutions within a population are encoded as chromosomes.

The GA implementation uses NSGA-II [2], and was chosen based on promising performance characteristics documented in [16]. It depends on implementation specific components used in every iteration, namely: Selectors selecting pairwise solutions. Crossover combining pairs of solutions. Mutators randomly modifying each solution. Fitness functions used for distinguishing good from bad solutions.

The termination criterion is set to a fixed number

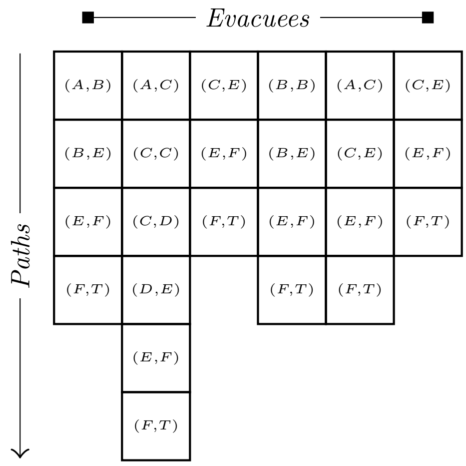


Figure 1: A example chromosome containing an evacuation routing plan.

of iterations. After termination, the best solution must be selected from the population. The population contains solutions that are optimized for one or more of the objectives. The final solution is selected by the super selector.

3.1 Encoding of Chromosomes

A chromosome contains path assignments for each evacuee as shown in Figure 1. Note that this is directly related to a selected search path s.

In this example the network includes nodes A-E, sink F and super sink T . The elements of each path, e.g. (A, B), are edges. Each column is a path assignment to either a person or a group which is initially located in the first node in the path. Edges are used in the path to support cases where multiple edges connect two nodes, such as if two oblong, parallel rooms are connected with two or more doors.

3.2 Genetic Operation

The GA starts by selecting a pair of parent chromosomes using binary tournament selection [2] and then performing a crossover using these.

3.2.1 Crossover

A multi-point crossover operator for recombining a pair of two-dimensional chromosomes has been implemented, by using one-point crossover once for each group represented in the chromosome. The crossover point in each parent is randomly selected among potential, valid crossover points. For a crossover point at node n_i to be considered valid, an edge with a target node n_i must exist within the corresponding path in both parent chromosomes. However, the common node does not need to be traversed at the same time.

If the crossover operation creates an invalid child, one of the parents is passed as child instead. Because different children for different lengths are created, the path may extend beyond the time-expansion. By ensuring that the initial population is valid and only passing valid solutions as children, this problem will not occur. It is important to note that this crossover operator can take two identical parents and still produce distinct children. Usually, a chromosome which is recombined with itself will produce children that are perfect copies of itself, and hence with no possible improvement. However, due to the way chromosomes are encoded and crossover is implemented, in this case a parent mating with itself has the possibility of producing offspring which are different and may be better than its parent. This effect arises because edges, which are reusable at different time steps, can occur several times in the same chromosome, which can lead to a single chromosome having multiple time-shifted crossover points with itself. Nevertheless, offspring that are identical to their single parent will still occur if the exact same crossover point is used for both of the parents. Reuse of edges allows for two things. Firstly, it allows waiting in a node by following the holdover edge two or more times in sequence. Secondly, it also occurs when paths are circular. Regardless, such solution paths will be evaluated by fitness functions and handled accordingly.

Note that in a time-expanded network, crossover breaks the sequential timing of the path. To fix the timing of a path, a repair function is applied after the crossover. This adjusts the time of an edge so that the sequential timing is kept.

3.2.2 Mutation

The one-dimensional mutation changes each path of each chromosome in the population by a predefined probability. The mutation generates a new random path starting from the same origin and ending at the super sink.

3.3 Fitness Functions

This section presents two fitness functions motivated by the evacuation criteria (see section 2) :survivability and grouping. As a direct consequence, the fitness functions are used from the problem formulation, namely: Survivability, $f(s)$ (see equation 1) and Grouping, $g(s)$ (see equation 2).

3.4 Super Selector

Unlike traditional GA, NSGA-II does not yield a single solution which can be considered the best one. This is an intended effect of using Pareto ranking. Instead, the solutions present in the first Pareto front are the set of the best solutions which the algorithm could find. Because the solutions with the same rank are mutually non-dominant NSGA-II makes no assumptions as to which, if any, of the objective functions are more important.

Therefore, a single solution must be extracted from the set of solutions yielded by NSGA-II. This can be done manually, which can be suitable in a decision-support system. However, automating the process is often preferable, which can be accomplished by adding a final processing step for the set of solutions NSGA-II yields. This can be realized by using a selection mechanism which is able to rank the solutions, for instance by combining the fitness values in some way. Here we use a prioritized fitness ranking approach.

Prioritizing works as follows. Starting with the highest ranked fitness measure, all solutions' value for it is compared. If one solution has a strictly lower value than all others, then that solution is selected. Otherwise, the set of solutions with the lowest value are compared again, this time on the next-highest ranking fitness measure. This continues until a solution has been found. If all objective functions have been processed in this way and more than one solution are still candidates, the tie is broken arbitrarily.

The objective functions we use are ranked in the following order:

- Endpoint capacity
- Survivability
- Passage congestion
- Room congestion
- Length

The ordering has been determined through informal reasoning. First, we definitely want every evacuee to be assigned to a lifeboat which has room for him or her. This is the highest ranked objective, seeing as failing to accomplish this is considered a hard failure (certain fatality). Second, we want to minimize the time spent in dangerous spaces, measured by survivability. Next comes congestion, which can influence actual survivability and cause evacuees to fail at following their assigned routes. Path length is selected last, because even thought it is desirable to have the shortest paths possible, it is less important than the other objectives.

4 Experimental results

This section provides results from running the algorithms NSGA-II, Djikstra and Random in a network based on the layout of a deck on a real ship, the MS Xpedition owned by Celebrity Cruises. This has in total 40 rooms and 4 lifeboats. For all experiments evacuees are randomly divided into groups and randomly spread out in the network. Survivability is distributed randomly between 0.8 and 1.0.

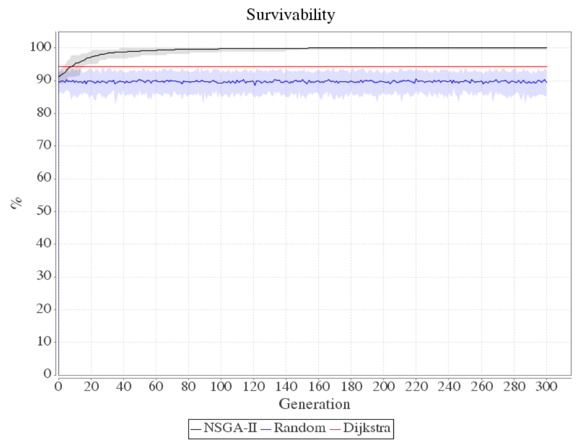
All results are aggregated averages of 50 iterations with the confidence of 95%.

Figure 2(a) and 2(b) show results from survivability for 20 evacuees in a situation with life boat capacity of 5, and random probability of surviving varying from 0.8 to 1.0. The evacuees were divided into random groups of 1-5.

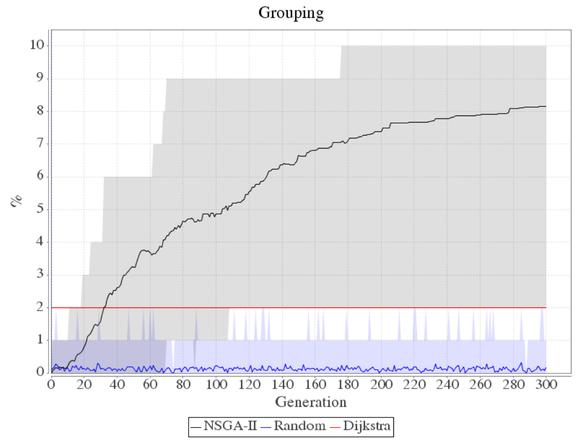
The results from Figure 2(a) indicates that NSGA-II finds the optimal solution (100% survivability), when there is no grouping, after 118 iterations. Djikstra is only able to survive 94% of the evacuees, while random is able to evacuate 90%.

The same experiment is present at Figure 2(b) and shows the grouping of each algorithm. It is noteworthy that the grouping continues to improve throughout the experiment.

A conclusion to be drawn from this is that the approach finds an optimal solution with respect to survivability. Further, without have any impact on the survivability, the approach continues to optimize on the grouping. This is in line with the intent (see section 1.1) that primarily optimises on survivability than continues optimisation on grouping.



(a) Survivability — $f(s)$



(b) Grouping — $g(s)$

Figure 2: Experiment results in a realistic ship network

4.0.1 Effects of Congestion Heuristic

In Figure 3(a) and , 3(b) the algorithms have been applied with the same parameters. However, in 3(a) NSGA-II was run without the congestion heuristics, while it was present in the experiment presented in 3(b). The effect this has can be seen clearly: When not optimizing for congestion, the criterion is neglected and increases as the genetic algorithm progresses. Congestion even approaches the value of the Random algorithm. Conversely, when congestion is optimized for, the genetic algorithm continually improves the congestion performance, albeit slowly. The results also show that the survivability improves correspondingly. However, for reasons of brevity, this graph is not presented here.

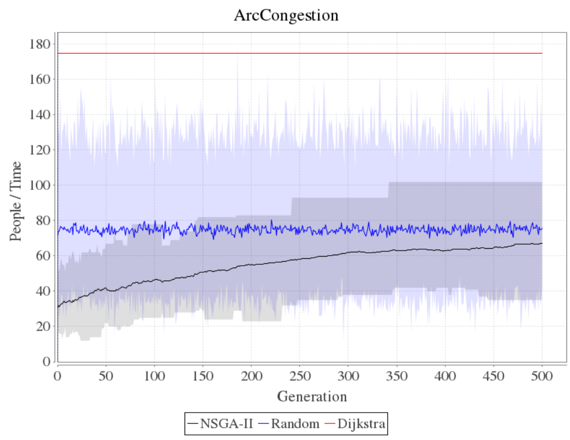
4.0.2 Extensive Experiments

Additional work has been done in both simple test networks and larger randomly generated graphs. The results from these graphs are similar to those presented in this chapter, with mostly negligible differences. The only difference worth mentioning is that randomly generated graphs yields a more difficult optimisation problem which decreases the performance of NSGA-II compared to Djikstra. Hence, NSGA-II result in a lower survivability than Djikstra

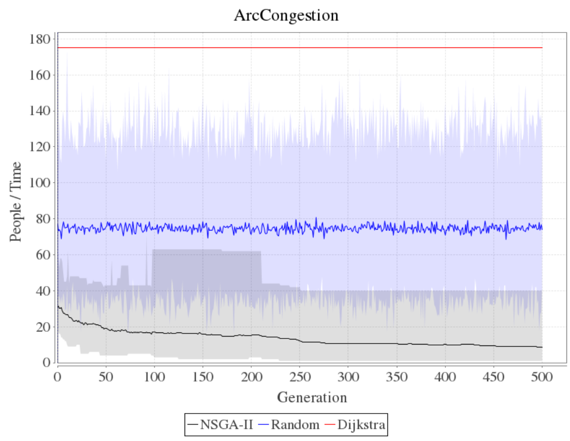
5 Conclusion

The problem of efficient evacuation can be viewed as an optimization problem, for which many techniques have been developed, including genetic algorithms.

NSGA-II is an adaption of the genetic algorithm framework which supports the preservation of diversity among candidate solutions by taking into



(a) Without congestion heuristics.



(b) With congestion heuristics.

Figure 3: Experiment results with and without congestion heuristics.

account Pareto indifference, meaning that no solution is strictly better than others. The technique presented in this paper bases itself on a

The technique we developed to select a solution from the multiple solutions returned by NSGA-II is a prioritized objective approach.

The results clearly show the potential genetic algorithms can have in evacuation planning. In fact, we found that in some simpler scenarios was able to find a solution for most of scenarios that outperformRandom and Djikstra's.

Our results also show that when the fitness functions becomes more complicated, such as considering congestion, the efficiency of the algorithm suffers. Hence, when the complexity increases more generations are needed.

5.1 Future Work

Future work includes making more specifically adapted genetic operators such as mutation operators which take into account the grouping aspect. The inherent complexity of the chromosome is likely a hurdle which needs to be overcome. Due to the way the chromosome is defined, very specific constraints are applied to it which limits the effectiveness of the genetic operators, compared to traditional genetic algorithms. The limitations are related to the way each part of the chromosome must be a valid path specification.

References

[1] Maged N Kamel Boulos, Bernd Resch, David N Crowley, John G Breslin, Gunho Sohn, Russ Burtner, William A Pike, Eduardo Jezierski, and Kuo-Yu Slayer Chuang. Crowdsourcing, citizen sensing and sensor web technologies for public and environmental health surveillance and crisis management: trends, ogc standards and application examples. *International journal of health geographics*, 10(1):67, 2011.

[2] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *Evolutionary Computation, IEEE Transactions on*, 6(2):182–197, 2002.

[3] Thomas E Drabek and David A McEntire. Emergent phenomena and the sociology of disaster: lessons, trends and opportunities from the research literature. *Disaster Prevention and Management*, 12(2):97–112, 2003.

[4] Lester R Ford and Delbert R Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8(3):399–404, 1956.

[5] Morten Goodwin, Ole-Christoffer Granmo, and Jaziar Radianti. Escape planning in realistic fire scenarios with ant colony optimisation. *Applied Intelligence*, 42(1):24–35, 2015.

[6] H.W. Hamacher and S.A. Tjandra. Mathematical modelling of evacuation problems–a state of the art. *Pedestrian and Evacuation Dynamics*, 2002:227–266, 2002.

[7] Sirisak Kongsomsaksakul, Chao Yang, and Anthony Chen. Shelter location-allocation model for flood evacuation planning. *Journal of the Eastern Asia Society for Transportation Studies*, 6(1):4237–4252, 2005.

[8] Nicholas D Lane, Emiliano Miluzzo, Hong Lu, Daniel Peebles, Tanzeem Choudhury, and Andrew T Campbell. A survey of mobile phone sensing. *Communications Magazine, IEEE*, 48(9):140–150, 2010.

[9] Gino J Lim, Shabnam Zangeneh, M Reza Baharnemati, and Tiravat Assavapokee. A capacitated network flow optimization approach for short notice evacuation planning. *European Journal of Operational Research*, 223(1):234–245, 2012.

[10] Anthony R Mawson. Understanding mass panic and other collective responses to threat and disaster. *Psychiatry: Interpersonal and biological processes*, 68(2):95–113, 2005.

[11] Adam J Pel, Michiel CJ Bliemer, and Serge P Hoogendoorn. A review on travel behaviour modelling in dynamic traffic simulation models for evacuations. *Transportation*, 39(1):97–123, 2012.

[12] Jaziar Radianti, Ole-Christoffer Granmo, Noureddine Bouhmala, Parvaneh Sarshar, Anis Yazidi, and Jose Gonzalez. Crowd models for emergency evacuation: A review targeting human-centered sensing. In *In Proceeding of 46 th Hawaii International Conference for System Sciences*, 2013.

[13] Mohammad Saadatseresht, Ali Mansourian, and Mohammad Taleai. Evacuation planning using multiobjective evolutionary optimization approach. *European Journal of Operational Research*, 198(1):305–314, 2009.

[14] Parvaneh Sarshar, Jaziar Radianti, and Jose J Gonzalez. On the impacts of utilizing smartphones on organizing rescue teams and evacuation procedures. In *Proceedings of the 11th International Conference on Information Systems for Crisis Response and Management, ISCRAM*, pages 24–27, 2015.

[15] James MacGregor Smith. Evacuation networks. In *Encyclopedia of optimization Ed. by Christodoulos A. Floudas and Panos M. Pardalos.*, pages 940–950. Springer, 2009.

[16] Eckart Zitzler, Kalyanmoy Deb, and Lothar Thiele. Comparison of multiobjective evolutionary algorithms: Empirical results. *Evolutionary computation*, 8(2):173–195, 2000.