Capacity Constrained Routing for Evacuation Planning

Qingsong Lu, Shashi Shekhar
Department of Computer Science and Engineering, University of Minnesota
200 Union St SE, Minneapolis, MN 55455, USA
E-mail: [lqingson, shekhar]@cs.umn.edu
WWW home page: http://www.cs.umn.edu/research/shashi-group

Abstract
Evacuation planning is critical for numerous important applications, e.g. emergency evacuation, disaster management and recovery, and homeland defense preparation. Efficient tools are needed to produce evacuation plans that identify routes and schedules to evacuate populations to safety in the event of catastrophes, natural disasters, and terrorist attacks. Current optimal methods suffer from computational complexity and may not scale up to large transportation networks. Current naive heuristic methods do not consider the capacity constraints of the evacuation network and may not produce feasible evacuation plans. In this paper, we model capacity as a time series and use a capacity constrained heuristic routing approach to solve the evacuation planning problem. We propose heuristic algorithm, namely Multiple-Route Capacity Constrained Planner (MRCCP) to incorporate capacity constraints of the routes. Experiments show that the proposed algorithms can produce close-to-optimal solution, and at the same time significantly reduce the computational cost compared with optimal algorithms. The experiments also show that our algorithms are scalable with respect to the number of evacuees and the size of the evacuation network.

1 Introduction
Evacuation planning is critical for numerous important applications, e.g. emergency evacuation, disaster management and recovery, and homeland defense preparation. Efficient tools are needed to produce evacuation plans that identify routes and schedules to evacuate populations to safety in the event of catastrophes, natural disasters, and terrorist attacks[1,2,3].

The current methods of evacuation planning can be divided into three categories, namely warning systems, assignment-simulation approach and route-schedule planning approach. Warning systems simply convey threat descriptions and the need for evacuation to the affected people via mass media communication methods. Such systems can have unanticipated effects on the evacuation process. For example, when Hurricane Andrew was approaching Florida in 1992, this approach caused tremendous traffic congestion, general confusion and chaos[4]. The assignment-simulation approach, such as DYNASMART[5], conducts stochastic simulation of traffic movements based on origin-destination demands and uses queuing methods for road capacity constraints. However, it may take long time to complete the simulation process for a large transportation network. The route-schedule planning approaches tend to use more sophisticated network flow and routing algorithms. EVACNET[6,7], Hoppe

---

1 This work is supported by the Army High Performance Computing Research Center (AHPCRC) under the auspices of the Department of the Army, Army Research Laboratory under contract number DAAD19-01-2-0014. The content does not necessarily reflect the position or policy of the government and no official endorsement should be inferred. AHPCRC and the Minnesota Supercomputer Institute provided access to computing facilities.

2 Corresponding author: Qingsong Lu, Email:lqingson@cs.umn.edu, Tel: 612-626-7703.
and Tardos[8] used linear programming methods to find optimal solution, which suffer from high computational overhead and may not be able to scale up to the large transportation networks characteristic of metropolitan areas. Heuristics routing algorithms can be used to find sub-optimal evacuation plan. However, current naive heuristic approaches only compute the shortest distance path from a source to the nearest destination without considering route capacity constraints and traffic from other sources. It cannot produce efficient plans when the number of evacuees is large and the road network is complex.

New heuristic approaches are needed to account for capacity constraints of the evacuation network. A capacity constrained routing approach reserves route capacities subject to capacity constraints in an order specified by heuristics. We propose a new heuristic algorithm, namely Multiple-Route Capacity Constrained Planner (MRCCP). This algorithm can assign multiple routes to groups of people from the same source based on an order prioritized by shortest travel time path lengths re-calculated in each iteration. Experimental results on a large dataset show that this approach produces close-to-optimal solutions with significantly reduced computational time compared to optimal solution algorithms. This algorithm is also scalable with respect to the total number of people to be evacuated and the size of the network. To the best of our knowledge, this is the first heuristic algorithm exploring capacity constrained routing for evacuation planning.

The rest of the paper is organized as follows. In Section 2, the problem formulation is provided and related concepts are illustrated by an example. Section 3 proposes two capacity constrained heuristic algorithms. The cost model of the algorithm is given in Section 4. In Section 5, we present the experimental results and the performance evaluation. We summarize our work and discuss future directions in Section 6.

2 Problem Formulation

The capacity constrained routing problem can be formulated as follows. Given a transportation network with capacity constraints, the initial number of people to be evacuated, their initial locations, and evacuation destinations, we need to produce evacuation route plans consisting of a set of origin-destination routes and a scheduling of people to be evacuated via the routes. The objective is to minimize the total time needed for evacuation. The scheduling of people onto the routes should observe the route capacity constraints. A secondary objective is to minimize the computational overhead of producing the evacuation plan.

We illustrate the problem formulation and a solution with the following example network, as shown in Figure 1. In this evacuation network, each node is shown by an ellipsis. Each node has two attributes: maximum node capacity and initial node occupancy. For example, at node N1, the maximum capacity is 50, which means this node can hold at most 50 people, while the initial occupancy is 10, which means there are initially 10 people at this node that are to be evacuated. In Figure 1, each edge, shown by an arrow, represents a link between two nodes. Each edge also has two attributes: maximum edge capacity and travel time. For example, at edge N4-N6, the maximum capacity is 5, which means at most 5 people can travel through this link within each time unit, while the travel time is 4, which means it takes 4 time unit to travel from node N4 to N6. This approach to model evacuation network with capacity to node-edge graph is similar to those presented in [7].

As shown in Figure 1, suppose we initially have 10 people at node N1, 5 at node N2, and 15 at node N8. The task is to compute an evacuation plan that evacuates the 30 people to the evacuation destination (N13 and N14) using the least amount of time.
Table 1 shows an evacuation plan. In the table, each row shows one group of people moving together during the evacuation with a group ID, number of people in this group, origin node, the start time, the evacuation route, and the exit time. Take node N8 for example, initially there are 15 people at N8. They are divided into 3 groups: Group A with 6 people, Group B with 6 people and Group C with 3 people. Group A starts at time 0, follows route N8-N10-N13 and reaches EXIT1(N13) at time 4. Group B starts at time 1, also follows route N8-N10-N13 and reaches EXIT2(N13) at time 5. Group C start at time 0, follows route N8-N11-N14 and reaches EXIT2(N14) at time 4. The procedure is similar for people from N1 and N2. The whole evacuation takes 16 time units since the last group of people (Group F and J) reaches the exit at time 16.

### 3 Capacity Constrained Routing Approach

We use a capacity constrained routing approach to conduct the evacuation planning. We model available edge capacity and available node capacity as a time series instead of a fixed number. A time series represents the available capacity at each time instant for a given edge or node. We propose an approach based on the extension of shortest path algorithms, e.g. Dijkstra’s algorithm, to account for route scheduling with capacity constraints. We proposed Multiple-Route Capacity Constrained Planner.
The MRCCP algorithm keeps iterating as long as there are still evacuees at any source node. In each iteration, the algorithm re-computes the earliest time route from any source to any destination taking the previous reservations and possible on-route waiting time into consideration. Then it reserves the capacity for this route in the current iteration. The detailed pseudo-code and algorithm description are as follows.

**Multiple-Route Capacity Constrained Planner (MRCCP)**

<table>
<thead>
<tr>
<th>Input:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1) $G(N, E)$: a graph $G$ with a set of nodes $N$ and a set of edges $E$;</td>
</tr>
<tr>
<td>Each node $n \in N$ has two properties:</td>
</tr>
<tr>
<td>Maximum Node Capacity(n) : non-negative integer</td>
</tr>
<tr>
<td>Initial Node Occupancy(n) : non-negative integer</td>
</tr>
<tr>
<td>Each edge $e \in E$ has two properties:</td>
</tr>
<tr>
<td>Maximum Edge Capacity(e) : non-negative integer</td>
</tr>
<tr>
<td>Travel Time(e) : non-negative integer</td>
</tr>
<tr>
<td>2) $S$: set of source nodes, $S \subseteq N$;</td>
</tr>
<tr>
<td>3) $D$: set of destination nodes, $D \subseteq N$;</td>
</tr>
<tr>
<td>Output: Evacuation plan</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method:</th>
</tr>
</thead>
<tbody>
<tr>
<td>while any source node $s \in S$ has evacuee do {</td>
</tr>
<tr>
<td>find route $R = n_0, n_1, \ldots, n_k$ &gt;= with earliest destination arrival time</td>
</tr>
<tr>
<td>among routes between all $s, d$ pairs, where $s \in S, d \in D, n_0 = s, n_k = d$;</td>
</tr>
<tr>
<td>$flow = \min (\text{number of evacuee still at source node } s,)$</td>
</tr>
<tr>
<td>Available Edge Capacity(all edges on route $R$),</td>
</tr>
<tr>
<td>Available Node Capacity(all nodes from $n_1$ to $n_k$ on route $R$),</td>
</tr>
<tr>
<td>};</td>
</tr>
<tr>
<td>for $i = 0$ to $k - 1$ do {</td>
</tr>
<tr>
<td>$t' = t + \text{Travel Time}(e_{n_i, n_{i+1}})$;</td>
</tr>
<tr>
<td>Available Edge Capacity($e_{n_i, n_{i+1}}$, $t'$) reduced by $flow$;</td>
</tr>
<tr>
<td>Available Node Capacity($n_{i+1}$, $t'$) reduced by $flow$;</td>
</tr>
<tr>
<td>$t = t'$;</td>
</tr>
<tr>
<td>}</td>
</tr>
<tr>
<td>Postprocess results and output evacuation plan;</td>
</tr>
</tbody>
</table>

The MRCCP algorithm keeps iterating as long as there are still evacuees at any source node (line 1). Each iteration starts with finding the route $R$ with the earliest destination arrival time from any sources node to any exit node based on the current available capacities (line 2). This is done by generalizing Dijkstra’s shortest path algorithm[9] to work with the time series capacities and edge travel time. Route $R$ is the route that reaches an exit in the least amount of time and at least one person can be sent to the exit through route $R$. For example, at the very first iteration, $R$ will be N8-N10-N13, which can reach N13 at time 4. The actual number of people that will travel through $R$ is the smallest number among the number of evacuees at the source node and the available capacities of each of the nodes and edges on route $R$ (line 3). Thus, in the example, this amount will be 6, which is the available edge capacity of N8-N10 at time 0.

The next step is to reserve capacities for the people on each node and edge of route $R$ (lines 4-9). The algorithm makes reservation for the 6 people by reducing the available capacity of each node and edge at the time point that they are at each node and edge. This means that available capacities are reduced by 6 for edge N8-N10 at time 0, for node N10 at time 3, and for edge N10-N13 at time 3. They finally arrive at N13(EXIT1) at time 4. Then, the algorithm goes back to line 2 for the next iteration.

The iteration terminates when the occupancy of all source nodes is reduced to zero, which means all evacuee have been sent to exits. Line 11 outputs the evacuation plan, as shown in Table 2.
We then provide the algebraic cost model for the computational cost of the proposed heuristic algorithm. We assume the total number of nodes in the graph is $n$, the number of source nodes is $n_s$, and the number of groups generated in the result evacuation plan is $n_g$. The MRCCP algorithm is an iterative approach. In each iteration, the route for one group of people is chosen and the capacities along the route are reserved. The total number of iterations is determined by the number of groups generated. In each iteration, the route with earliest destination arrival time from each source node to any exit node is re-computed with the cost of $O(n_s \times n \log n)$. Reservation is made for the node and edge capacities along the chosen route with the cost of $O(n)$. The cost model of the MRCCP algorithm is given as follows:

$$Cost_{MRCCP} = O((n_s \times n \log n + n) \times n_g)$$

### 5 Performance Evaluation

#### 5.1 Experiment Design

The purpose of the experiments is to compare the quality of solution and the computational cost of MRCCP with that of EVACNET which produces optimal solution. First, a test dataset is an evacuation network characterized by its route capacities and its size (number of nodes and edges) is chosen or generated. Next, a generator is used to generate the initial state of the evacuation by populating the network with a distribution model to assign people to source nodes. The initial state will be converted to EVACNET input format to produce optimal solution via EVACNET and converted to node-edge graph format to evaluate the proposed MRCCP algorithm. The solution qualities and algorithm performance will be analyzed in analysis module.

The generator produces initial states by varying source node ratio and occupancy ratio from 10% to 100%. The experiment was conducted on a workstation with Intel Pentium III 1.2GHz CPU, 256MB RAM and Windows 2000 Professional operating system.

The initial state generator distributes $P_n$ people to $S_n$ randomly chosen source nodes. The source node ratio is defined as $S_n / (total \ number \ of \ nodes)$ and the occupancy ratio is defined as $P_n / (total \ capacity \ of \ all \ nodes)$.

We want to answer two questions: (1) How does people distribution affect the performance and solution quality of the algorithms? (2) Are the algorithms scalable with respect to the number of people to be evacuated?

#### 5.2 Experiment Results
Experiment 1: Effect of People Distribution

The purpose of the first experiment is to evaluate how the people distribution affects the quality of the solution and the performance of the algorithms. We fixed the occupancy ratio and varied the source node ratio to observe the quality of the solution and the running time of the proposed algorithm and EVACNET.

The experiment was done with fixed occupancy ratio from 10% to 100% of total capacity. Here we present the experiment results with occupancy ratio fixed at 30% and source node ratio varying from 30% to 100% which shows a typical result of all test cases. Figure 2 shows the total evacuation time given by the two algorithms and Figure 3 shows their running time.

![Figure 2. Quality of Solution With Respect to Source Node Ratio](image)

![Figure 3. Running Time With Respect to Source Node Ratio](image)

As seen in Figure 2, at each source node ratio, MRCCP produces solution with total evacuation time that is within 10% longer than optimal solution produced by EVACNET. The quality of solution of MPCCP is not affected by the distribution of people when the total number of people is fixed. In Figure 3, we can see that the running time of EVACNET grows much faster than the running time of MRCCP when source node ratio increases.
This experiment shows: (1) MRCCP produces close to optimal solution (less than 10% longer than optimal) with less than half of running time of EVACNET. (2) The distribution of people does not affect the performance of MRCCP when total number people is fixed.

Experiment 2: Scalability with Respect to Occupancy Ratio

In this experiment, we evaluated the performance of the algorithms when the source node ratio is fixed and the occupancy ratio is increasing.

![Figure 4. Quality of Solution With Respect to Occupancy Ratio](image1)

![Figure 5. Running Time With Respect to Occupancy Ratio](image2)

Figure 4 and Figure 5 show the total evacuation time and the running time of the 2 algorithms when the source node ratio is fixed at 70% and occupancy ratio varies from 10% to 70% which is a typical case among all test cases.
As seen in Figure 4, compared with the optimal solution by EVACNET, solution quality of MRCCP still remains within 10% longer than optimal solution. In Figure 5, the running time of EVACNET grows significantly when occupancy ratio grows, while running time of MRCCP remains less than half of EVACNET and only grows linearly.

This experiment shows that MRCCP is scalable with respect to number of people.

5.3 A Real Scenario

We also conducted experiments using a real evacuation scenario, as shown in Figure 6. The Monticello nuclear power plant is about 40 miles to the northwest of the Twin Cities. Hand-drafted evacuation route plan was developed to evacuate the affected population to a high school in case of accidents or terrorist attacks. However, the plan did not consider the capacity of the road networks and put high loads on two highways. In addition, it did not consider the actual population of the nearby cities.

We conducted experiment using our MRCCP algorithm. The experiment was done using the same evacuation network in Figure 6 and Census 2000 population data for each affected city (yellow circles in Figure 6).

As can be seen in Figure 6, given the evacuation road network and the population data, our algorithm gives much better evacuation route plan by selecting shorter paths to reduce total evacuation time and utilizing richer routes (green routes near destination) to reduce congestion.

![Figure 6. Result Routes Overlay of Monticello Power Plant Evacuation Planning](image)

6 Conclusion and Future Work

In this paper, we proposed and evaluated a new heuristic algorithm of capacity constrained routing approach. Cost models and experimental evaluations using real dataset are presented. The proposed
MRCCR algorithm produces close-to-optimal solution and at the time significantly reduces the running time. The algorithm is scalable to the number of evacuees. We also conducted experiments using large size transportation network, e.g. Twin Cities Metro Area roadmaps, to test the scalability of the new algorithm. The results show that our algorithm is scalable with respect to the size of the transportation network.

Our approach can also be used to determine the optimal transportation network configuration, e.g. one-way directions and reversible lanes, to maximize the traffic flow rate during an evacuation. We plan to extend our approach to address this problem.

Acknowledgements

We are particularly grateful to Spatial Database Group members for their helpful comments and valuable discussions.

References