A Simulation-based Approach for Large-scale Evacuation Planning

Kazi Ashik Islam*, Madhav Marathe*, Henning Mortveit† Samarth Swarup, Anil Vullikanti*,
Biocomplexity Institute and Initiative,
*Department of Computer Science,
†Department of Engineering Systems and Environment,
University of Virginia
{*ki5hd,marathe,henning.mortveit,swarup,vsakumar*}@*virginia.edu*

Abstract—**Evacuation planning methods aim to design routes and schedules to relocate people to safety in the event of natural or man-made disasters. The primary goal is to minimize casualties which often requires the evacuation process to be completed as soon as possible. In this paper, we present QueST, an agent-based discrete event queuing network simulation system, and STEERS, an iterative routing algorithm that uses QueST for designing and evaluating large scale evacuation plans in terms of total egress time and congestion/bottlenecks occurring during evacuation. We use the Houston Metropolitan Area, which consists of nine US counties and spans an area of 9,444 square miles as a case study, and compare the performance of STEERS with two existing route planning methods. We find that STEERS is either better or comparable to these methods in terms of total evacuation time and congestion faced by the evacuees. We also analyze the large volume of data generated by the simulation process to gain insights about the scenarios arising from following the evacuation routes prescribed by these methods.**

I. Introduction

Evacuation models are designed to ensure the safe relocation of people from an area under threat of natural or man-made disaster. Examples of such situations are hurricanes, floods, tsunamis, wildfires, terrorist attacks, hazardous chemical spills, etc. Any of these scenarios can pose a serious threat to human lives and evacuation is often the best option to minimize casualties. Large scale evacuation planning has proven to be essential in the past. A prime example of this is Hurricane Rita which made landfall in September 2005 and affected areas in Florida, Mississippi, Louisiana, Texas, Arkansas, Missouri, and Illinois. Before its landfall, approximately 2.5 million people were evacuated from the Texas coastal area [1]. It was one of the largest urban evacuations in US history. This coastal area is still prone to hurricanes every year. Even at the time of writing this article, Hurricane Laura in Aug 2020, a category 4 hurricane on the verge of becoming category 5, is about to make landfall near the Louisiana and Texas coastal area [2]. Evaluating the effectiveness of an evacuation plan is an important part of the planning process. Simulation techniques have been used extensively for this purpose [3]–[7]. However, simulating a hurricane evacuation scenario in the above mentioned areas is quite challenging due to its sheer scale. The Greater Houston area alone consists of nine US counties with a population of about six million (Census 2010). A simulator will have to work with a large road network. Also, the simulation process will generate a large volume of data containing the location traces of the evacuees during the evacuation. This can amount to hundreds of gigabytes data. These data need to be analyzed to understand the pros and cons of different evacuation plans. To the best of our knowledge, the existing literature lacks a system that has simulated an evacuation planning method at this scale (in terms of network size, number of evacuees, realistic representation of network congestion and complex evacuation scenarios).

As our first contribution, we present an agent-based discrete event queuing network simulation system named 'Queuing Simulation of Traffic' (QueST) that is capable of simulating evacuations at the scale mentioned above. The system represents a road network as a queuing network and vehicles travelling through the network as agents. Given the agents' initial locations, destinations, evacuation routes and the road network, QueST simulates the movement of all the agents until everyone reaches their destination. To capture the effect of congestion, we use state-dependent queues. We also use publicly available real-world traffic data to tune a traffic model that can estimate the effective speed on a road at a given traffic density. QueST uses this model to estimate travel time on roads. The system is also inherently capable of capturing scenarios where vehicles cannot enter a road due to that road being full.

As our second contribution, we choose the Houston metropolitan area as our case study and use QueST to design and then evaluate evacuation plans for this area. The metropolitan has a total area of 9,444 square miles and is inhabited by about six million people. The area has been affected by major hurricanes (e.g. Hurricane Katrina & Rita (2005), Ike (2008), Harvey (2017), Laura & Marco (2020)). An illustration of the evacuation scenario for this area is shown in Figure 1. The red dots represent the initial locations of the evacuees and the green circles denote the safe zones. We have used two existing routing methods to determine the evacuation routes: Shortest Path routing and CASPER [4]. In addition, we present a new routing method named 'Shortest Time Estimate-based Evacuation Route Selection' or STEERS, as our third contribution. In STEERS, we use

978-1-7281-6251-5/20/$31.00 ©2020 IEEE

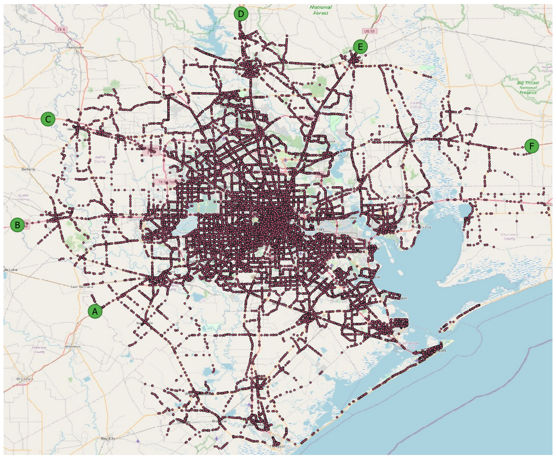


Fig. 1: Evacuation Scenario in Houston

the data generated by the simulator to iteratively find better routes for the evacuees. Using the routes from these methods, we have simulate the evacuation process. The simulation runs provide us the location traces of the evacuees and the states of the roads during the evacuation. We analyze these data for calculating metrics such as the evacuation completion time, egress time of each evacuee, congestion level on the roads, blocking time of the evacuees on different roads etc. We also present a comparative analysis of the three routing methods.

The rest of the article is organized as follows, in section II, we review existing works on queuing networks, evacuation modeling and simulation models for evacuation. In section III, we present the details of our simulation model. Section IV briefly describes two existing routing algorithms and STEERS. Details of the data we have used and of the traffic model we have tuned are presented in section V. Section VI contains the details of the simulation experiments together with related analysis and results. Finally, we conclude the article by mentioning our future plans with this system in section VII.

II. Literature Review

Research on evacuation planning can be categorized into three main categories: Macroscopic, Microscopic, and Mesoscopic models. Macroscopic models approach evacuation planning as an optimization problem and do not consider microscopic level details such as individual behavior or decision of selecting egress routes. Hamacher and Tjandra first presented such a mathematical model for evacuation where they modeled the problem as a dynamic network flow problem on a capacity limited network [8]. To solve it as a static flow problem, they introduced the idea of Time Expanded Graphs (TEG). Although TEG based methods provide optimal results, they are computationally very expensive. To overcome this, Lu, George and Shekhar proposed the 'Capacity Constrained Route Planner' (CCRP), a heuristic algorithm that generates both schedules and routes for the evacuees and does not need time expanded graphs [9]. Later the method was improved by Kim, George and Shekhar [10] to have lower run-time complexity for application to large transportation networks.

One limitation of the CCRP algorithm is that it assumes travel time on a specific road to be a constant. As a solution, Shahabi and Wilson [4] presented 'Capacity-Aware Shortest Path Evacuation Routing' (CASPER). It is a heuristic method for evacuation routing that considers the effect of congestion on the effective speed on a road. CASPER utilizes traffic models for this purpose and can work with different traffic models. For validation, the authors compared the estimated evacuation time of CASPER with the simulated evacuation time of a flocking simulation model [11]. However, they only did this validation for a small evacuation scenario. Later, the authors extended their work for dynamic environments [12] .

All the proposed methods mentioned so far used a macroscopic approach for finding egress routes. Microscopic simulation models such as car following models [13]–[16], cellular automata models [17], and multi-agent models have been used to analyze vehicular traffic flow. In these models, each vehicle is considered as a separate object and therefore their behavior and interaction with other objects is modeled at a high level of detail. However, these simulation models tend to be computationally expensive. To address the resulting scalability issue, Gawron presented a queuing network model [18] for traffic micro-simulation where each road in the road network is represented as a queue. The model used a fixed-increment time progression approach. It was later implemented in MATSim [19], an open-source framework for transport simulation. Another queue-based discrete event simulation model with the next-event time progression approach [20] was later added to MATSim. Both of these are microscopic models. In contrast, we present a mesoscopic simulation model, where we combine macroscopic supply (e.g., link capacities and link performance in terms of the speed-density relationship) with microscopic demand (e.g., individual drivers).

Most of the recent works on evacuation planning and simulation use car following models for mobility simulation. Na and Banerjee [5] presented an agent-based discrete event simulation framework, embedded within a GIS module to simulate evacuation plans designed for no-notice disasters. The authors used the car following model of Yang and Koutsopoulos [13] for simulating the agents' movements. In their experiments, they simulated an evacuation of the city of San Francisco. Hasan and Hentenryck [6] presented several zone-based evacuation planning algorithms where each zone is assigned a schedule for its evacuation. They used the SUMO simulator [16], which is based on a car following model, to evaluate their proposed algorithms. The same simulator was used by Chen, Shafi and Chen [7] where the authors presented a simulation pipeline for understanding the challenges of emergency evacuation through case studies. Two case studies were presented in this work. To the best of our knowledge, the largest evacuation scenario simulated in the literature worked with approximately 1 million evacuees in the coastal city of Padang, Indonesia [3]. The authors of this work used MATSim for traffic simulation. In comparison, our study area is more than thirty times larger and also contains about six times the number of evacuees.

III. Queuing Network Simulation Model

The QueST system involves two parts: (1) building a queuing network from a given road network (Section III-A), and (2) the simulation of the evacuation process (Section III-B).

A. Constructing the Queuing Network

Let's consider the sample road network in Figure 2a where the nodes denote the junctions and the edges denote the roads. Each edge has the following properties: length of the road (e_{length}), number of lanes (e_{lanes}), and number of vehicles that fit on it per lane and per unit length (e_K).

Given these values, we can calculate the maximum number of vehicles that can simultaneously exist on the road. We call it the capacity of the road e_C, given by: $e_C = e_K * e_{length} * e_{lanes}$. To transform the road network into a queuing network, we consider each edge as a queue that has e_C number of servers and no buffer or waiting space. The service pattern of each queue is state-dependent, i.e., the service time on a queue (which corresponds to the travel time on its associated edge) is dependent on the number of vehicles present in the queue. Once we have transformed all the edges into queues, we connect the pairs of queues whose corresponding edges are incident edges in the road network. This completes the construction of the queuing network. The edge to queue mapping for the sample network is shown in Figure 2b. The corresponding queuing network is shown in Figure 2c.

B. Simulation Process

To describe the simulation process of the evacuation, we use our sample network again. The evacuees start evacuation from node 1 and 2 and they need to be evacuated to node 4. In the corresponding queuing network (Figure 2c), vehicles will enter the system through queue A and B, and exit it through queue C. The two evacuation routes are shown in Figure 2d. Evacuees from nodes 1 and 2 follow routes 1 and 2 respectively. The vehicles arrive at the initial nodes (i.e., nodes 1 and 2 in our example) at a given rate. The inter-arrival time of the vehicles can be modeled using any distribution, e.g., an exponential distribution. If the edge a vehicle wants to enter has available capacity, it can enter the edge immediately. Otherwise, it will have to wait. Each vehicle follows a predetermined route and it does so in a *hold-and-wait* manner. As a concrete example, let's say a vehicle is currently on edge (1, 3) and it wants to enter edge (3, 4) next. The vehicle will wait for a free space to be available on edge (3, 4) and until that time it will hold space on edge (1, 3). A detailed algorithmic description of a single vehicle's traversing behavior is presented in Algorithm 1. The *hold-and-wait* behavior is described in lines 4-6. The travel time of a vehicle on an edge is calculated using the *calculate_service_time* function. We can use different formulations for calculating this value. Constant, linear, exponential, power and logistic models [4], [21] are some of the formulations used in the literature. These models use the capacity and current number of vehicles on the road to calculate the travel time.

Vehicles that request access to the same edge are granted access in a first-come-first-served manner. In case multiple vehicles request access at the same time, the tie is broken by granting them access in an arbitrary order.

Algorithm 1: Vehicle Route Traversal Process

```
   Input: Evacuation Route (a sequence of edges): route
1  prev_edge ← null
2  for edge in route do
3      edge_request ← request for access to edge
4      while edge_request not successful do
           // wait till space is available on edge
5          wait
       // access to edge acquired, now release
6      prev_edge Release prev_edge
7      service_time ← calculate_service_time(edge)
8      while elapsed time on edge < service_time do
9          wait
10     prev_edge ← edge
```

IV. Evacuation Routing Algorithms

In this section, we describe the routing algorithms that we have used in our experiments. First, we present the statement of the evacuation routing problem. Then, we describe two existing routing algorithms, namely Shortest Path Routing and CASPER [4]. Finally, we describe our iterative routing algorithm, STEERS.

A. Problem Statement

Given a set of source locations $S = \{s_1, s_2, ..., s_n\}$, a set of safe zones $D = \{d_1, d_2, ..., d_m\}$, number of evacuees w_i at each source location s_i $(1 \leq i \leq n)$, and a road network $G = (V, E)$, the goal of the evacuation routing problem is to: (1) find a set of routes

$$R = \{r_i \mid r_i = route(s_i, d_j), s_i \in S, d_j \in D\}$$

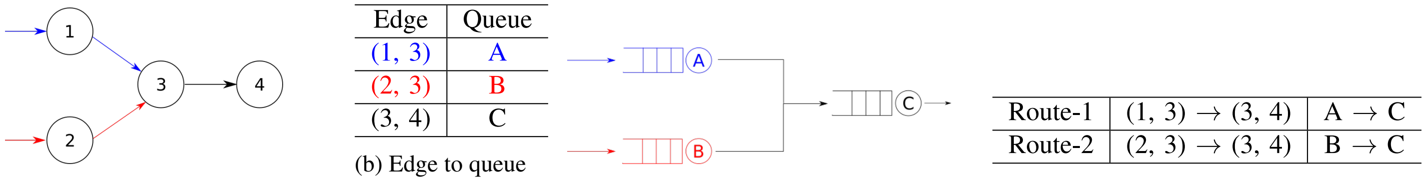
that contains a route from each source node in S to any safe zone in D, and (2) determine a schedule for moving the evacuees on the chosen routes. Different objectives have been considered for the choice of routes and schedules in the literature. Most routing algorithms primarily try to optimize the evacuation completion time, which is the time when every evacuee has reached a safe zone [4], [8]–[10]. Multi-objective formulations are also available [22]. However, even the simplest setting of minimizing the maximum delay (referred to as the *makespan*), for constant link capacity, is NP-complete.

B. Two Existing Routing Algorithms

We describe two prior algorithms, which are used as baselines for our experiments: Shortest Path Routing and CASPER.

1) *Shortest Path Routing (SP)*: In this method, each source location is paired with its respective nearest (by road) safe zone. The shortest path from each source location to their paired safe zone is used as the evacuation route for that source.

2) *CASPER*: It is a heuristic method that considers the effect of congestion during the route calculation. The method uses Dijkstra's algorithm as a subroutine but with a modified cost function. This function utilizes a traffic model to estimate the delay caused by congestion. For details about this method, we refer to the work of Shahabi and Wilson [4].



(a) Sample Road Network

Edge	Queue
(1, 3)	A
(2, 3)	B
(3, 4)	C

(b) Edge to queue mapping

(c) Corresponding Queuing Network

Route-1	(1, 3) → (3, 4)	A → C
Route-2	(2, 3) → (3, 4)	B → C

(d) Evacuation Routes

Fig. 2: Queuing Network Construction

```
Algorithm 2: Algorithm for STEERS
   Input: Percentage of routes to update in each iteration: $p$
   Output: Evacuation Routes
 1 Set number of evacuees to one for each source location
 2 $current\_routes \leftarrow$ shortest path routes
 3 do
 4     Run simulation using $current\_routes$
 5     Calculate avg. travel time on each edge from simulation results
 6     Set the edge weights to the calculated travel times
 7     $new\_routes \leftarrow$ Re-calculated shortest path routes
 8     $candidate\_source\_locations \leftarrow$ randomly choose $p\%$ of the source locations
 9     for $src$ in $candidate\_source\_locations$ do
10         if estimated cost on new route for $src$ < simulated cost on current route of $src$ then
11             update the $current\_route$ of $src$ to the $new\ route$
12 while there is an estimated improvement in travel time on at least one route ;
13 return $current\_routes$
```

Data	**Source**	**Usage**
Road Network Data of Houston	HERE Maps	To build the road network graph and the queuing network
Census Data (2010) for Houston Metropolitan Area	U.S. Census Bureau	To determine number of evacuees and their initial locations
Real-world Traffic Data	Georgia Department of Transportation (GDOT)	To calculate the parameters of our traffic model

TABLE I: Summary of Data used

Property	**Value**
Number of Nodes	129,950
Number of Edges	182,084
Average Degree	1.4

TABLE II: Properties of the Directed Graph Representing the Road Network of the Houston Metropolitan Area

C. Shortest Time Estimate-based Evacuation Route Selection (STEERS)

We present an iterative routing algorithm that uses the output from our simulation tool to find good routes. The main idea for the algorithm is as follows: when we simulate the movement of the vehicles through the road network, the simulator knows the state of all the edges in the network at all times. This provides us with valuable information about how congested the roads become and the average travel time on those roads. We use this information to come up with better routes. The outline of the algorithm is shown in Algorithm 2.

The algorithm first sets the number of evacuees at each source location to one and the initial routes to the shortest paths (lines 1-2). A simulation is then run with this setting (line 4). From the results of the simulation, we calculate the average travel time on each edge (line 5). We then calculate new routes for each of the source locations using these values as edge weights (line 7). However, we only update a certain percentage ($p\%$) of routes and do so only if there is an estimated improvement in travel time (lines 9-11). The intuition here is that evacuees from these locations are trying to change their route (and possibly their destination too) to decrease their egress time. This completes one iteration of the algorithm. The iterations continue until no better route can be found for any of the source locations (line 12). The algorithm can also be terminated after a certain number of iterations or when the maximum estimated improvement among all the sources is smaller than a certain threshold. The latest set of routes are returned as the final evacuation routes. Our motivation for designing the algorithm in this way is to find an equilibrium where no evacuee can change their route and do better than their current route.

V. Data Sources and Calibration

We have used data from multiple sources for our simulation experiments. In this section, we provide details about the sources of these data and how we used them in our system. A summary is provided in Table I.

A. Road Network Data

We have used HERE maps data [23] to construct the road network of our study area. The data contains details including: name of the roads, their start and end nodes, geometry, function class (e.g arterial, collector), number of lanes, speed limit, etc. We have used this information to (1) build a directed graph representing the road network of the whole study area—Table II shows some properties of this graph; (2) construct the queuing network (section III-A); and (3) design evacuation routes (section IV).

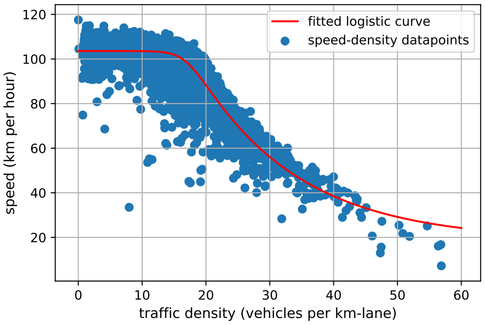


Fig. 3: Density-vs-Speed Data Points (from one sensor) and Fitted Logistic Curve

B. Census Data

We have collected the census data of 2010 from the US Census Bureau, which contains all the census block groups in the Houston Metropolitan area (33,641 in total). Each block group is a polygon and contains the number of housing units in it and the number of people living in those houses. The total number of people in this area is 5,861,300. We determined the centroid of each block group and then used these points as the initial locations (i.e. sources) of the evacuees. The number of evacuees in each source location is set equal to the number of people living in the corresponding block group. Before simulation, we snapped the location of the sources to their nearest nodes in the road network. The snapped locations are shown as red dots in Figure 1. As safe zones, we chose six locations on major highways coming out of the metropolitan area (green circles in Figure 1). The intuition is that when evacuees reach any of these places, they are out of danger and can follow the highway to their preferred location.

C. Traffic Data

To get a realistic estimation of evacuation time, we need to estimate travel time on roads in different congestion situations. The effective speed on a road is heavily influenced by the number of vehicles travelling on it. In traffic flow theory, the metric traffic density (number of vehicles per lane and per unit length) is generally used to estimate effective speed. We study the correlation between traffic density and speed by analyzing traffic data (vehicle count and speed) from Georgia Department of Transportation (GDOT[1]). We have extracted one year of traffic data (year 2019) from 233 roadside sensors. The roads monitored by these sensors have a total of 986 lanes and are also of varying function classes. We calculated the traffic density vs. speed data points from the raw, per hour traffic count and speed data. A visualization of the data points from one sensor, monitoring a road of function class one, is shown in Figure 3 (blue dots).

Our observation from several such plots is that the speed-vs-density relationship seems to fit a reverse 'S' shaped curve. Similar observation was reported by Wang et al. [21]. We have thus used their proposed logistic model to estimate speed from density. A logistic curve fitting the observed data is shown in Figure 3 (red curve). It has five parameters: free flow speed (v_f), average speed at stop and go condition (v_b), density (k_t) at which the traffic flow transitions from free flow to congested flow, scale parameter (θ_1) and shape parameter (θ_2). The speed-vs-density relationship according to the logistic model is shown in equation 1 (here Θ represents all five parameters).

$$v(k, \Theta) = v_b + \frac{v_f - v_b}{\left(1 + e^{\frac{k-k_t}{\theta_1}}\right)^{\theta_2}} \tag{1}$$

We fit a logistic curve to data from each lane of a road. Then we grouped the curves based on the function class of the roads. We observed clear differences among the optimized parameter values of different function class roads. However, intra-class variations also exist. We choose the median values for each function class as the representative value of that class. For the free flow speed parameter, we use the speed limit value when available. During simulation, depending on the function class of a road, we use the appropriate parameter values to estimate speed from traffic density.

VI. Experiment Results

In this section we provide details about the experiments we have performed. First, we describe the experiment settings including some details about the implementation of our system. Then we provide detailed analysis of the simulation results.

A. Experiment Settings

We have simulated the evacuation at two scales:
1) *Small Scale*: Here, we assume that there is only one evacuee at each source location. So, effectively we simulate the evacuation of 33,641 vehicles.
2) *Large Scale*: At this scale, we evacuate everyone (about six million people) from the study area. We assume that at each source location, on average every three people (the average number of people per household in the US) use one car. So during simulation, about two million cars travel through the network from the sources to the safe zones.

In experiments with both scales, we assume that evacuation from all the sources start at the same time. However, at each source location, every car leaves at two second time intervals. We consider evacuation routes from three routing algorithms: Shortest Path routing, CASPER, and STEERS. Also, we consider evacuation by private vehicles only.

We have used the process-based discrete-event simulation framework Simpy[2] in our implementation. The entire system is implemented in Python. The simulations and subsequent analyses were performed in a High Performance Computing cluster, with 128GB RAM and 16 CPU cores allocated to our task. Our simulator is currently a single threaded program. The sixteen cores are allocated to help speed up the analyses of the simulation results.

[1] https://gdottrafficdata.drakewell.com/publicmultinodemap.asp

[2] https://simpy.readthedocs.io/en/latest/

Routing Algorithm	Route Length (km)	Travel Time (hour)	Destination Safe Zone
SP	173.03	4.05	A
CASPER	141.85	2.54	A
STEERS	88.09	2.26	A

TABLE III: Details of Last Evacuee's route

	A	B	C	D	E	F
SP	**16787**	1742	2920	4095	5530	2567
CASPER	**9830**	3388	3532	6506	7211	3174
STEERS	**8763**	2558	3377	5993	6213	6737

TABLE IV: Number of Evacuees Assigned to Safe Zones

B. Analysis of the Results

We have organized our analyses by addressing some key questions regarding the evacuation process. Here, we present each of these questions and the necessary analyses to answer them. At first, we answer the questions using the results from the small-scale simulation. Later, we inspect the large-scale simulation results and present our findings.

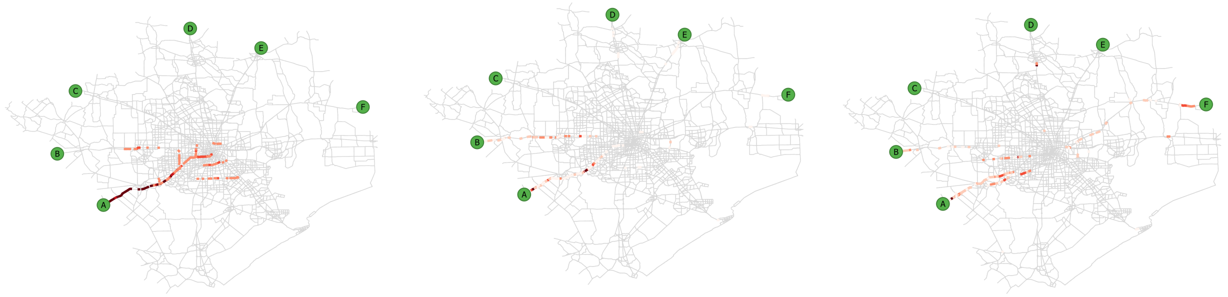
Evacuees from which source location reach a safe zone last? The length, travel time, and destination of the routes taken by the last evacuee are shown in Table III. We can see from this table that in the small scale scenario, STEERS routes evacuate everyone faster (i.e., in 2.26 hours) than the other two methods. Also, we see that the last evacuee's destination is safe zone 'A' for all three methods. To understand why that is the case, we look at the number of evacuees assigned to each safe zone in Table IV. In all three methods, safe zone 'A' is assigned the largest number of evacuees. This happens because it is spatially closer to more source locations than other safe zones (Figure 1). As more travelers head towards 'A', congestion occurs on the roads leading to this destination. However, as seen from Table IV, CASPER and STEERS try to re-distribute the evacuees to other safe zones to avoid this situation.

Which road segments are the most congested? To identify the road segments that are most congested throughout the evacuation, we record the traffic density on every road during the simulation. Figure 4 shows the top 300 roads which have the highest mean traffic density (throughout the evacuation) for all three routing methods. The figure shows how SP routes cause high congestion on routes towards safe zone 'A'. CASPER and STEERS, on the other hand, successfully spread out the congestion over other roads by doing a more balanced assignment of the evacuees to the safe zones.

How long does each evacuee experience congested traffic flow? Congested traffic flow occurs when vehicles are at standstill or are travelling at a considerably lower speed than the free flow speed on a road. Naturally, we want to avoid congested traffic flow. For our analysis, we consider traffic flow of an evacuee to be congested when the traveler is at standstill or is travelling at less than 30% of the free flow speed of the road segment. To quantify the evacuee's experience, we calculate for each evacuee, the ratio of travel time under congested traffic flow to total travel time. We show these values (in percentage) in Figure 5 using a box-plot. The figure shows that on SP routes, half of the evacuees experience congested traffic flow for more than 67.7% (the median) of their total evacuation time. CASPER shows improvement in this respect with a corresponding median value of 45.1%. STEERS further improves this with the lowest median value of 26.7%. Also, the variance of this same metric is found to be lower for STEERS (323.36) than SP (1004.15) and CASPER (644.96). The results imply that the experience (regarding congestion) of different evacuees is more similar in STEERS compared to SP and CASPER. In this sense, STEERS produces a more fair assignment of routes as compared to SP and CASPER.

What is the rate of evacuation throughout the evacuation process? Evacuation completion time only tells us when all of the evacuees have reached a safe zone. However, it is often important to evacuate each person as quickly as possible. In Figure 6, the percentage of people evacuated with time is shown for each of the three routing methods. The slope of the tangent at any point of the curve denotes the evacuation rate at that point of time. We have added some randomness in the simulation runs by (i) letting each evacuee leave his source location in a five minute time window from the start of the simulation, and by (ii) adding some randomness to the estimated travel time given by the logistic traffic model. We have done this to account for the inherent uncertainty regarding when evacuees start their evacuation and the actual travel time on the road. This is also important to see how the results for the same routing method change because of the added randomness. With this randomness, we ran five simulations for each of the three routing methods. Summary of the evacuation completion times from these simulation runs are shown in Figure 7. We see an interesting pattern from Figures 6 and 7. For the same routing method, evacuees reach their safe zones at different times in different simulation runs. This is apparent by examining Figure 7. However, the curves showing the evacuation rate in different simulation runs (Figure 6), are very similar, i.e., the red, green, and blue curves are very tightly grouped. This implies that the added randomness affects when a specific person reaches a safe zone. But the overall evacuation rate at different points of time remains similar. This is true for routes prescribed by all the three routing methods. In general, the green curves are the steepest ascending in Figure 6. This suggests that routes found by STEERS results in a higher evacuation rate as compared to routes found by SP or CASPER.

How do the routing methods perform in the large scale evacuation scenario? So far, we have seen how the different routing methods perform comparatively in the small scale scenario. Now, we present a comparative analysis of the methods in the large scale scenario. Before going into details, we mention that in the large scale scenario, the same set of routes from SP and STEERS are used. Ideally in STEERS, we would run the simulations iteratively with all evacuees



(a) SP

(b) CASPER

(c) STEERS

Fig. 4: Most Congested Roads.

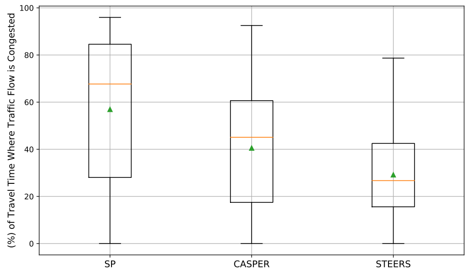


Fig. 5: Percentage of travel time where traffic flow is congested

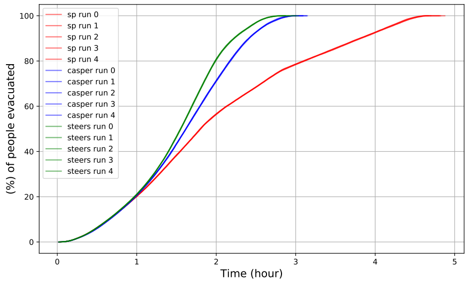


Fig. 6: Evacuation rate in small scale scenario

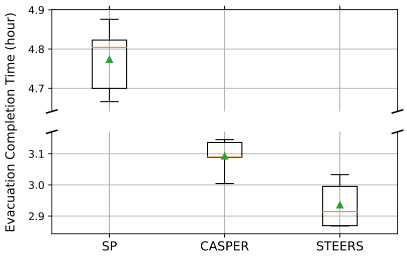


Fig. 7: Evacuation completion times in five simulation runs.

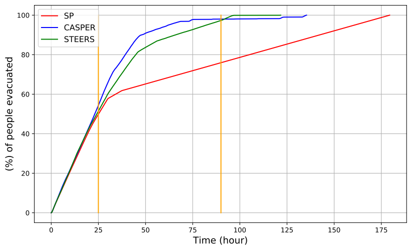


Fig. 8: Evacuation rate in large scale scenario

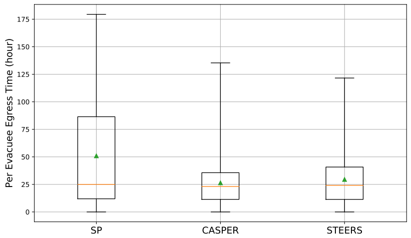


Fig. 9: Per evacuee egress time in large scale scenario

instead of just one evacuee per source location. However, each iteration of a large scale simulation takes about 1.5-2 days. That is why in this work, we decided to inspect how the routes generated from iterative small scale simulation perform in the large scale scenario. In contrast, CASPER generates different evacuation routes based on the actual number of evacuees at the source locations and can do so without requiring any extra resource. For this reason, we have generated new CASPER routes using the actual number of evacuees.

The percentage of people evacuated with time is shown in Figure 8. We can see from the plot that for all the three methods the evacuation completion time is more than 100 hours. However, about half of the people are evacuated in a day. Following the SP routes results in the longest time to complete the evacuation. Surprisingly, STEERS routes show a lower evacuation completion time (122 hours) than CASPER

(135 hours). We can see that between hour 25 and 90, the blue curve is steeper than the green one. This means evacuation rate in this time window is higher for CASPER than STEERS. Near hour 90, the blue curve becomes almost flat, implying that the evacuation rate is very low there. We can look at Figure 9 to understand why this is happening. This figure shows the egress time of different evacuees using a box-plot. We can see that the third quartile for CASPER (35.62 hour) is less than the third quartile for STEERS (40.76 hour). This means that CASPER routes have evacuated more people than STEERS in 40 hours. This is also confirmed in Figure 8. However, a small number of evacuees in CASPER have high egress time that even exceeds the maximum egress time in STEERS. These are the people who are arriving at the safe zones in the flat region of the blue curve. Due to the better evacuation completion time of STEERS, but the higher evacuation rate of CASPER, there is no clear winner between STEERS and CASPER in the large scale scenario. However, as mentioned earlier, running STEERS with all the evacuees is expected to capture the evacuation scenario in a more realistic manner. Further studies will be carried out to understand the relative trade-offs between the two methods.

VII. Conclusion and Future Plans

We present a discrete event queuing simulation system (QueST) and simulate an evacuation in the Houston Metropolitan area. In the simulator, we use a traffic model that we tuned using real-world traffic data. We also present an iterative algorithm (STEERS) that can leverage the data generated by the simulator to find evacuation routes. Through experiments, we show in multiple ways that the performance of STEERS is either better or comparable to two existing routing algorithms. In future works, we plan to add more features to our simulator, such as zone based scheduling, contraflow lanes, etc.

VIII. Acknowledgment

This work was partially supported by the NIH Grant 1R01GM109718; NSF Grants: BIG DATA IIS-1633028, OAC-1916805, Expeditions in Computing CCF-1918656, RAPID OAC-2027541, CMMI-1832587; NASA Applied Sciences Program Grant #80NSSC18K1594; and DARPA under Contract No. HR0011-19-C-0096.

References

[1] S. K. Carpender, P. H. Campbell, B. J. Quiram, J. Frances, and J. J. Artzberger, "Urban evacuations and rural america: lessons learned from hurricane rita," *Public Health Reports*, vol. 121, no. 6, pp. 775–779, 2006.

[2] "Hurricane Laura becomes "extremely dangerous" Category 4 storm," 2020. [Online]. Available: https://www.cbsnews.com/live-updates/hurricane-laura-storm-track-path-forecast-today-2020-08-26/

[3] G. Lämmel, H. Klüpfel, and K. Nagel, "The MATSim network flow model for traffic simulation adapted to large-scale emergency egress and an application to the evacuation of the indonesian city of padang in case of a tsunami warning," in *Pedestrian Behavior*. Emerald Group Publishing Limited, Nov. 2009, pp. 245–265. [Online]. Available: https://doi.org/10.1108/9781848557512-011

[4] K. Shahabi and J. P. Wilson, "CASPER: Intelligent capacity-aware evacuation routing," *Computers, Environment and Urban Systems*, vol. 46, pp. 12–24, Jul. 2014. [Online]. Available: https://doi.org/10.1016/j.compenvurbsys.2014.03.004

[5] H. S. Na and A. Banerjee, "Agent-based discrete-event simulation model for no-notice natural disaster evacuation planning," *Computers & Industrial Engineering*, vol. 129, pp. 44–55, Mar. 2019. [Online]. Available: https://doi.org/10.1016/j.cie.2019.01.022

[6] M. H. Hasan and P. V. Hentenryck, "Large-scale zone-based evacuation planning: Models, algorithms, and evaluation," 2020.

[7] Y. Chen, S. Y. Shafi, and Y. fan Chen, "Simulation pipeline for traffic evacuation in urban areas and emergency traffic management policy improvements through case studies," 2020.

[8] H. Hamacher and S. Tjandra, "Mathematical modeling of evacuation problems: A state of the art," *Pedestrian and Evacuation Dynamics*, vol. 2002, 01 2002.

[9] Q. Lu, B. George, and S. Shekhar, "Capacity constrained routing algorithms for evacuation planning: A summary of results," in *Proceedings of the 9th International Conference on Advances in Spatial and Temporal Databases*, ser. SSTD'05. Berlin, Heidelberg: Springer-Verlag, 2005, pp. 291–307. [Online]. Available: http://dx.doi.org/10.1007/11535331_17

[10] S. Kim, B. George, and S. Shekhar, "Evacuation route planning: Scalable heuristics," in *Proceedings of the 15th Annual ACM International Symposium on Advances in Geographic Information Systems*, ser. GIS '07. New York, NY, USA: ACM, 2007, pp. 20:1–20:8. [Online]. Available: http://doi.acm.org/10.1145/1341012.1341039

[11] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques - SIGGRAPH '87*. ACM Press, 1987. [Online]. Available: https://doi.org/10.1145/37401.37406

[12] K. Shahabi and J. P. Wilson, "Scalable evacuation routing in a dynamic environment," *Computers, Environment and Urban Systems*, vol. 67, pp. 29–40, Jan. 2018. [Online]. Available: https://doi.org/10.1016/j.compenvurbsys.2017.08.011

[13] Q. Yang and H. N. Koutsopoulos, "A microscopic traffic simulator for evaluation of dynamic traffic management systems," *Transportation Research Part C: Emerging Technologies*, vol. 4, no. 3, pp. 113–129, Jun. 1996. [Online]. Available: https://doi.org/10.1016/s0968-090x(96)00006-x

[14] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Physical Review E*, vol. 62, no. 2, pp. 1805–1824, Aug. 2000. [Online]. Available: https://doi.org/10.1103/physreve.62.1805

[15] G. F. Newell, "A simplified car-following theory: A lower order model," *Transportation Research Part B: Methodological*, vol. 36, no. 3, pp. 195–205, March 2002. [Online]. Available: https://ideas.repec.org/a/eee/transb/v36y2002i3p195-205.html

[16] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO - Simulation of Urban MObility," 2012.

[17] K. Nagel and M. Schreckenberg, "A cellular automaton model for freeway traffic," *Journal de Physique I*, vol. 2, no. 12, pp. 2221–2229, Dec. 1992. [Online]. Available: https://doi.org/10.1051/jp1:1992277

[18] C. Gawron, "An Iterative Algorithm to Determine the Dynamic User Equilibrium in a Traffic Simulation Model," *International Journal of Modern Physics C*, vol. 09, no. 03, pp. 393–407, may 1998. [Online]. Available: https://www.worldscientific.com/doi/abs/10.1142/S0129183198000303

[19] A. Horni, K. Nagel, and K. W. Axhausen, *The Multi-Agent Transport Simulation MATSim*. London, GBR: Ubiquity Press, 2016.

[20] R. A. Waraich, D. Charypar, M. Balmer, and K. W. Axhausen, "Performance improvements for large-scale traffic simulation in MATSim," in *Computational Approaches for Urban Environments*. Springer International Publishing, Nov. 2014, pp. 211–233. [Online]. Available: https://doi.org/10.1007/978-3-319-11469-9_9

[21] H. Wang, J. Li, Q.-Y. Chen, and D. Ni, "Logistic modeling of the equilibrium speed–density relationship," *Transportation Research Part A: Policy and Practice*, vol. 45, no. 6, pp. 554–566, Jul. 2011. [Online]. Available: https://doi.org/10.1016/j.tra.2011.03.010

[22] A. Stepanov and J. M. Smith, "Multi-objective evacuation routing in transportation networks," *European Journal of Operational Research*, vol. 198, no. 2, pp. 435–446, Oct. 2009. [Online]. Available: https://doi.org/10.1016/j.ejor.2008.08.025

[23] "HERE Premium Streets Data set for the U.S." 2020. [Online]. Available: https://www.here.com/