EvacSys: A Cloud-based Service for Emergency Evacuation

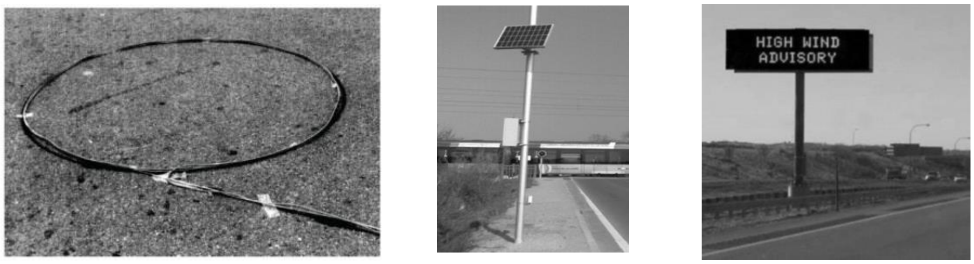
Osman Khalid, Muhammad Usman Shahid Khan, Ying Huang,
Samee U. Khan, *Senior Member IEEE,* and Albert Zomaya, *Fellow IEEE*

Abstract—Natural or man-made disasters wreak havoc, whether they are floods, earthquakes, tornados, hurricanes, or wild fires. One of the major challenges in emergency situations is to guide people through safer routes away from the disaster site on the basis of the available information. Data from large number of sources, such as road side sensory units, emergency vehicles, satellite imagery must be processed at real-time to compute the appropriate routes for evacuees. However, to process the huge volumes of sensory data at real-time also requires higher computational resources. For many years, cloud computing has been well-established as a most reliable solution to meet higher data and computational demands. In this magazine article, we discuss the architecture of a scalable cloud-based emergency evacuation service, namely *EvacSys.* The service utilizes the power of cloud-computing to process large volumes of real-time sensory data gathered during disaster and computes appropriate routes for evacuees with priority given to emergency vehicles. A case study is also presented on a real city transportation network to test the service.

Index Terms—Route recommendation service, emergency evacuations, traffic modeling, scalable service

1 Introduction

Throughout history, all communities have faced various disasters, both natural and human-induced. According to a study released by International Monetary Fund (IMF), since 2010, more than 700 disasters have been registered worldwide, affecting more than 450 million people [1]. Moreover, the report also indicated that since the 1990s, annual damages have risen from an average of US $20 billion to about $100 billion, and the upward trend in such losses is expected to continue. Protecting life against any disaster is a multifaceted problem, confronted by citizens, as well as, local, state, and federal governments. One of the major aspects of evacuation planning is the selection of safer routes, which is of critical importance for saving lives during a disaster situation. Generally, the transportation departments plan road networks by considering the traffic demands at busy hours, e.g., at beginning or end of office timings. However, it is difficult to predict the traffic volumes during disaster scenarios, as large groups of people try to evacuate the site as early as possible. The unavailability of proper evacuation plans may lead to road congestion and traffic jams that may also result in loss of human lives. For instance, it was reported that during the Katrina and Rita hurricanes, lack of proper evacuation plans resulted in heavy traffic jams on interstates. In another incident, 25 people lost their lives in just half hour when fires erupted in Oakland Hills California in 1991, as people got trapped on the road due to congestion. Recently, a 20 hours traffic jam was observed in Atlanta, USA in 2014 during a winter storm, as the road network was incapable of handling road congestion due to snow and accidents. Therefore, it is of utmost importance to design route recommendation plans and services that guide people towards safer routes with least congestion and risk during an emergency evacuation.



(a) (b) (c)

Fig. 1. Examples ITS sensors: (a) induction loop sensor, (b) weather sensor, and (c) alerts

In an effort to perform real-time traffic monitoring, since 1990s, the U.S. transportation planning departments have continued to engage in the design of Intelligent Transportation Systems (ITS) [2] that primarily consists of: **(a)** a sensing system, **(b)** a communication system, **(c)** Road Side Units (RSUs), made up of different types of sensors (Fig. 1), e.g., induction loop sensors, weather sensors, video cameras, radio-frequency identification (RFID) readers, traffic signal control system, and **(d)** notification system that includes car navigation and alerts. The ITS equipment is usually deployed at various intersections and roads within a city environment to acquire daily traffic information, resulting in a citywide network of ITS [3], [4].

During a disaster scenario, ITS can play a pivotal role in sensing the information pertaining to road conditions, extent of damage to a road, traffic volume, and amount of congestion found on a road [5]. Such information must be processed on the real-time to compute safer evacuation routes for the evacuees. However, with the massive volumes of data collected by large number of ITS equipment, the real-time processing of such information is a daunting task and requires large-scale computational and storage resources. Cloud Computing is an emerging and innovative platform that provides on-demand scalability of computing and storage resources to the end-users. In disaster scenarios, cloud computing can be practically applied to process massive volumes of ITS-based sensory information, and to compute at the real-time a set of safer and least congested routes for evacuees. As the incoming disaster information could be through multiple sources, the timely fusion of such information is critically important and requires higher resources, which is practical through cloud computing [14].

In this magazine article, we present an overview of how cloud computing can be effectively utilized for route recommendation in emergency evacuation scenarios. We discuss the architecture of a cloud-based route recommendation service, namely *EvacSys* that leverages the power of cloud computing to compute routes for evacuees that are least congested, at minimum risk, and of shortest-length. The service obtains real-time data from the ITS and computes routes for evacuees with priority given to the emergency vehicles including police cars, ambulances, and fire brigade trucks. To perform the empirical testing of the service, we present a case study in which people are evacuated away from the disaster site through routes computed by the service. To make the service scalable, we utilized parallel computing using Message Passing Interface (MPI) on compute cluster in a cloud environment. We incorporated various stochastic factors that usually affect the evacuations in disaster scenarios, so that the presented case study closely matches with the realistic scenarios.

The remainder of this article is organized as follows. The major components of the cloud-based *EvacSys* service are described in Section 2. In Section 3, we present the design and model of the service. Simulation results are discussed in Section 4. In Section 5, we present conclusions and future work.

2 SERVICE ARCHITECTURE

The generic architecture of *EvacSys* service is presented in Fig. 2. The vehicles are categorized into two categories: Evacuees' vehicles and Special Emergency Vehicles (later termed as emergency vehicles throughout in the paper) such as ambulances, fire department trucks, and police vehicles. The other entities involved are roads, disaster site, and safety locations. The collection of traffic/road condition information is performed by

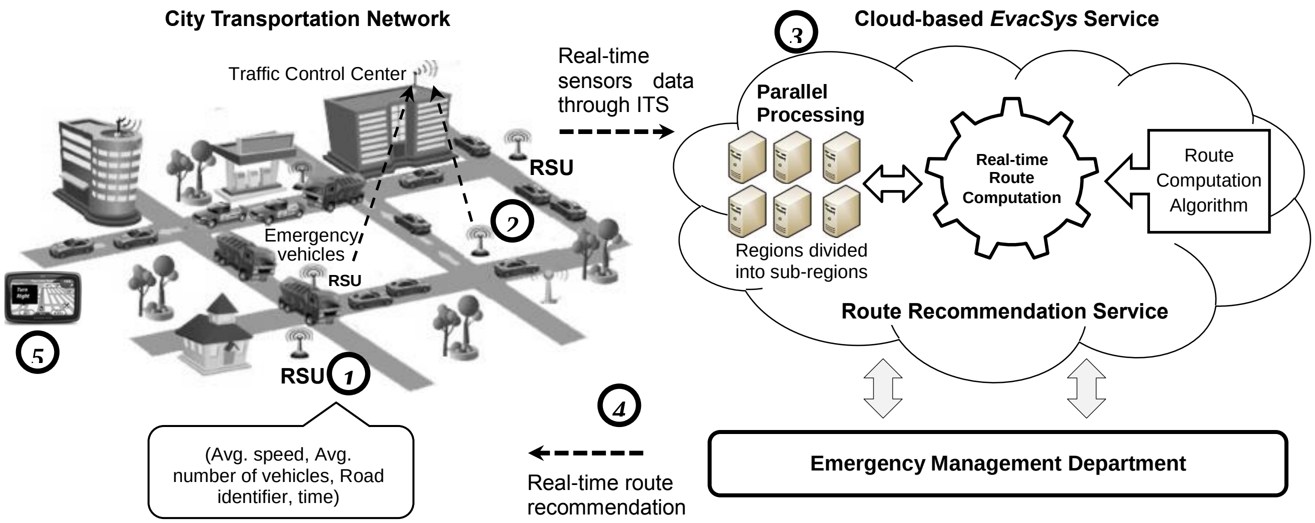


Fig. 2. A top level architecture of the cloud-based *EvacSys* service

RSUs mostly deployed on intersections throughout city transportation network. The RSUs capture road and disaster related information, such as average speed of vehicles, average number of vehicles, road condition, and flood level, etc (Step 1). Using any of the different communication technologies, such as satellite communications, 3G, UMTS, WLAN, GSM/GPRS, or any other backbone networks/gateways [14] [15], the vehicles' locations and sensory data is streamed towards the Traffic Control Center, where *EvacSys* service is running on a cloud-based parallel computing cluster (Step 2). The service maintains the city map at which the route cost from each intersection to each of the safety locations is computed and stored in a database. With the new incoming sensory information, the route costs are continuously updated in the database. The cluster performs the parallel processing of various sub-regions of the city map and calculates the congestion on each road segment (Step 3). The route computation algorithm utilizes the current traffic and hazard information to find out a subset of routes that have the capacity to allow maximum traffic flow with minimum delay. The computed set of routes is communicated to the emergency management department to take appropriate decisions during evacuations, as well as to the emergency vehicles and evacuees for traffic guidance. The emergency vehicles approaching towards congested road links are given priority over evacuees' vehicles, and all the vehicles are warned in advance about the congestion. To prevent congestion, alternate routes are provided on the navigation devices, ITS warning sign-boards, or other means of communication (as discussed above), such as radio or smart phones. The next section presents design and model of *EvacSys* service.

3 System Design

We consider a transportation network with three major entities: (a) disaster site, (b) road network, (c) safety shelters. Disaster site acts as group of *source* nodes from where evacuees will evacuate, whereas safety shelters are *destination* points for evacuees. The road network is the set of *routes* whose subset is to be travelled by the population during evacuation. The objective is to minimize the evacuation time under a given set of constraints, e.g. maximum capacity of road, available capacity, congestion, and risk factor. As the disaster evolves over time, the links that were not affected (or congested) initially may become affected at a later time, e.g., in case of flood, fire, or hurricane. Therefore, we consider the transportation network as a dynamic network and traffic is modeled at macro-scale level, where the vehicles are carrying evacuees from sources to destinations, with priority given to emergency vehicles. During evacuation, the vehicles take dynamic route choices based on the route cost (discussed later). The route cost computation requires speed and accuracy that can only be achieved through parallel implementation of route cost algorithm, which we achieved through parallel computing on cloud-based cluster. Dynamic re-routing of vehicles closely depicts a real life scenario in which the vehicles can be informed about road conditions in advance by the help of navigation systems, updates on radio, or road side ITS equipment [5]. More importantly, capturing time-evolving stochastic variations in the road network during modeling will only make the model more realistic. We discuss a few stochastic factors in the subsequent text.

3.1 Departure Rate and Departure Time

Departure rate indicates the average number of vehicles departing from homes within a disaster affected area. The number of evacuating vehicles is a stochastic parameter that depends on the number of evacuees residing in a house. Usually, in such scenarios, the evacuees departures are independent of each other and makes a discrete count 0, 1, 2, …, n. We can relate such phenomenon with querying theory, where the calls arrivals are independent, and make up a discrete count, and is generally represented by $Poisson(\lambda)$ distribution. Therefore, we also model the departure rate with Poisson distribution, which also gives additional benefit in terms of implementation, as it is having just a single parameter (λ) that needs to be varied during simulations. Moreover, as we are considering a macro-level traffic model, we are not taking into account the types of housings or details specific to the structure of buildings. Generally, when an evacuation warning is issued, few people comply immediately and leave initially, then the number of people evacuating reaches a peak value, and gradually drops down as most of the population has already been evacuated. We model such a behavior in terms of departure times with a response curve expressed with probability density function of Weibull distribution [6]. The Weibull distribution produces a response curve that has departure rate proportional to a power of time. Compared to the Poisson and Uniform distributions, the Weibull distribution more closely depicts the evacuee's compliance behavior due to the asymmetric nature of the curve.

3.2 Safety Shelter Selection

When the evacuees are guided away from the disaster site, there must be some destination points towards which the evacuees travel on safer routes. The route cost is calculated between the current location (intersection) at which the evacuee is present, and the target destination shelter that is nearest to the evacuee away from disaster. At the time of disaster, since a large number of people are moving towards safety shelters, there is a quite possibility that a shelter becomes overcrowded after some time. Therefore, while computing route cost, the model also considers the current space available in a safety shelter. If a shelter has no capacity, alternate shelter is considered for the route cost computation. Our model assumes that by the use of sensor technology, the real-time information is available about how much space is left in a given shelter.

3.3 Route Selection

The *EvacSys* service allows individuals to take decisions about route selection at road intersections. As a disaster has tendency to expand outwards from the epicenter, such as in the case of wild fire, tornado, and floods, the roads that are not affected by disaster yet may get hit by the disaster at a later time. To accommodate such behavior in simulations, the roads are randomly destroyed in disaster affected areas. We assume that the information about road damage is provided at real-time by the ITS sensors. Moreover, when large number of vehicles enter a road segment, the route cost also increases due to the increase in congestion, as vehicles' speed is reduced during heavy traffic.

3.3.1 Route Cost

For various major intersections, the *EvacSys* service maintains a route cost table that contains least travel costs from the intersection to each of the safety destinations. Route cost is the time a vehicle will take in traversing a route to reach the destination shelter. Route cost is calculated from the following information about a road segment: **(a)** maximum traffic flow capacity, **(b)** amount of congestion, **(c)** free flow, and **(d)** maximum possible speed of vehicles on the road segment.

Algorithm 1. Route Recommendation

Input: City transportation network graph G, and a set of safe destinations.

Output: Least cost route towards a preferred destination.

```
Split graph G into a number of sub-regions
destinations ← GetDestinations()
for all regions in parallel do
   for all intersections within the region do
     if dest ∈ destinations is within the same region then
        if there is an emergency vehicle on the way of evacuee vehicle then
          Compute secondmin(route cost) to the dest
        else
          Compute min(route cost) to the dest
        end if
     else
        if there is an emergency vehicle on the way of evacuee vehicle then
          Compute secondmin(route cost) to the boundary point
        else
          Compute min(route cost) to the boundary point
        end if
     end if
   end for
   for each vehicle v in a sub-region, that has not reached the shelter do
      pref_destination ← PreferredDestination(vehicle_location)
   end for
   Recommend the minimum cost route to evacuee towards preferred destination
end for
```

- *Maximum traffic flow capacity:* It is the maximum number of vehicles that can traverse a road segment without congestion. It is defined as the product of road segment length and number of lanes divided by average length of vehicles.
- *Amount of congestion:* Congestion is the ratio of number of vehicles travelling through the road segment to the maximum traffic flow capacity.
- *Free flow:* It is the product of maximum traffic flow capacity and maximum allowed speed limit of the road segment.
- *Maximum possible speed of vehicles:* It is the ratio of free flow to number of vehicles travelling through road segment.

Finally, we compute the cost of road segment k as a ratio of road segment length to maximum possible speed of vehicles through the road segment. The total cost of a route between an intersection and the destination shelter is the sum of the costs of individual road segments.

In normal situations, inter-vehicular distance depends on speed of vehicles. More the speed of vehicles, higher would be the inter-vehicular distance. However, we assume that during the time of disaster, it is almost impossible for the vehicles to maintain appropriate inter-vehicular distance, because, evacuees are trying to save their lives.

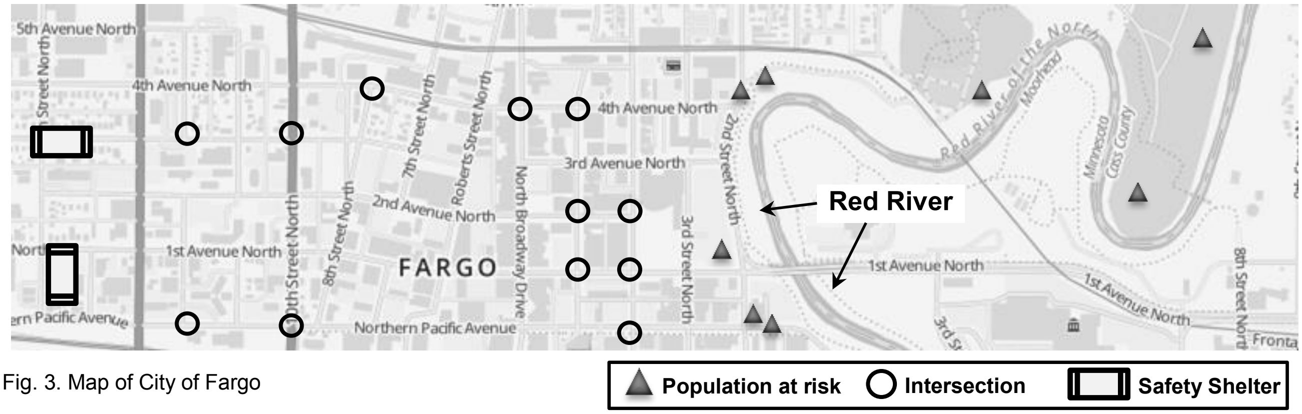


Fig. 3. Map of City of Fargo

3.3.2 Route Computation Algorithm

Fig. 3 depicts a portion of the map of City of Fargo that we considered as a case study in this paper. For designing and testing of our proposed *EvacSys* service, we considered the flooding of Red River in North Dakota State, USA [7]. Examples of large-scale evacuations for major Red River flooding events happened in 1826, 1897, 1950, 1997, 2009, and 2011 and there is risk of more flooding in future [12]. To model our system, we obtained the data, including road capacities, traffic volumes, speed limits, and historic flood affected areas, from the City of Fargo [8] and North Dakota Department of Transportation (NDOT) [9].

The three main items marked on the map are disaster risk areas, intersections, and safety shelters [10][11]. During the evacuation, the residents in affected areas flee towards safety shelters, and are guided about the routes on intersections. We denote the road network with a graph notation: $G = (V, E)$, where V is the set of vertices representing intersections, and E is the set of links that represent roads. We split the graph into several sub-graphs and each sub-graph is processed on a separate node in parallel. We denote a sub-graph by $H^{(r)}$, $r = 1, 2, 3, \dots, n$, such that $H^{(1)} \cup H^{(2)} \cup H^{(3)} \cup \dots \cup H^{(n)} = G$. Between any two regions $H^{(a)}$ and $H^{(b)}$, we define a set of overlapping points as *boundary points* B_{ab}. The boundary points are the intersection nodes that are common to both the regions, such that $B_{ab} = H^{(a)} \cap H^{(b)}$. Suppose, evacuees in region $H^{(a)}$ are recommended a shelter that is located in the region $H^{(r)}$, where $r \neq a$. As a first step, the evacuees are routed towards a boundary point $p_i \in \{B_{ak} = H^{(a)} \cap H^{(k)}\}$ that has the minimum congestion and risk at the current time interval (determined through ITS). Here, $H^{(k)}$ is the region adjacent to $H^{(a)}$, and located on the least cost path towards the target shelter. On reaching the boundary point p_i using least cost paths, as the next step, vehicles are routed within the same region $H^{(k)}$ based on route cost computations. If k=r, then the vehicles have reached the desired region, where the target shelter is located. Otherwise, the aforementioned procedure will be repeated to further route the vehicles towards new boundary point.

The Algorithm 1 computes, and recommends least cost routes to the evacuees during disaster. The city graph is split into several sub-regions, and a set of available shelters is initialized (Line 1 – Line 2). Each sub-region graph is given as input to a separate cloud-based cluster node, so that all the regions are processed in parallel (Line 3). Within each sub-graph/sub-region, the route cost from all the intersections to each of the destinations is updated in the route cost tables maintained by each intersection. Here, two cases may arise. If destination shelter is within the same region, and there is an emergency vehicle on the way of evacuating vehicle, then the evacuee vehicle is guided on the second minimum cost route towards destination to avoid any possible congestion for emergency vehicle, otherwise, it is guided towards minimum cost route (Line 4 – Line 11). In second case, if the destination is not within same sub-region, the vehicles are guided towards a boundary point, again prioritizing emergency vehicle (Line 12 – Line 18). As illustrated above, the boundary points are a set of overlapping nodes that are common in any two adjacent regions. It is possible that multiple regions are adjacent to one region. Therefore, boundary point will be the one that is located on the least cost (and safest) path towards the target destination. With the help of ITS, the model checks the congestion/risk status of all the boundary points and selects those that are least congested and safer. When a vehicle crosses a boundary point and enters into the newer region, a *handover* takes place in which the older region hands over the vehicle information to the newer region. If the target destination is in same region, the vehicle is routed towards destination; otherwise, it is routed towards new boundary point. If the roads towards a selected destination become inaccessible, or a shelter becomes overcrowded, a new preferred destination will be selected for the evacuee (Line 19 – Line 21).

4 PERFORMANCE EVALUATION

4.1 Simulation Settings

To perform the empirical testing of the proposed service, we have carried out experiments on our internal cloud-based HPC cluster setup. For our simulations, the total number of evacuees is considered to be 108,000.

If the given map is converted to a graph representation, then the graph has 2,800 vertices with 7,370 edges. To more closely depict a disaster's behavior in terms of damaging road networks, we randomly mark roads as destroyed, beginning with the roads geographically near to a disaster site. As the simulation time proceeds, the road destruction is expanded outwards from disaster site to mimic damage caused by the floods. Most of the existing baseline approaches are based on multi-objective optimizations that find a set of optimal solutions in the form of routes from a static graph [13]. However, in our scenario the network is dynamic, with real-time varying parameter. Therefore, it is impossible to compare our scheme with any of the existing approaches that are based on multi-objective optimization. Moreover, to the best of our knowledge, there is no existing work that provides route recommendations for evacuees with preference given to emergency vehicles as we presented in this article. Therefore, for comparisons, we developed two of our baseline approaches, namely: **(a)** Dedicated and **(b)** Shortest-Path. In the Dedicated scheme, evacuees follow only those roads that were specifically dedicated for evacuations by Fargo department of transportation [8], [9]. Alternatively, Shortest-Path approach allows evacuees to take the shortest routes from disaster site towards safety shelter. The following performance metrics are considered for evaluations**:**

- *Average travel time:* It is the average of travel times of all the evacuees.
- *Average congestion:* As defined in Subsection 3.3.1, it is average of congestion experienced by a road segment at a given time interval.
- *Average evacuation time:* It is the average time spent between the start of evacuation and when the last person evacuates the affected area.

4.2 Simulation Results

In this subsection, we present the simulation results for the emergency evacuation scenario considered in this magazine article. Due to space limitations, we have included only a subset of results from the large number of simulations that we conducted using the real map of the city. For each data point, the simulation is repeated 10 times to obtain the statistical significance of the results. The number of emergency vehicles is taken as 20 in the simulations (unless stated otherwise).

Fig. 4(a) depicts average travel time of the evacuees by varying the departure rate (λ) of vehicles. As reflected from Fig. 4(a), with the increase in departure rate, the average travel time of *Dedicated* and *Shortest-Path* turns out to be higher than the *EvacSys*. This is because these approaches do not take into account the current traffic flow rate on roads. As people tend to adopt the shortest route for evacuations, the roads become congested and road's traffic flow rate is dropped. Same is the reason for *Dedicated* approach as arrival of too many vehicles on the pre-defined limited set of roads results in congestion of such roads. Fig. 4(b) indicates the

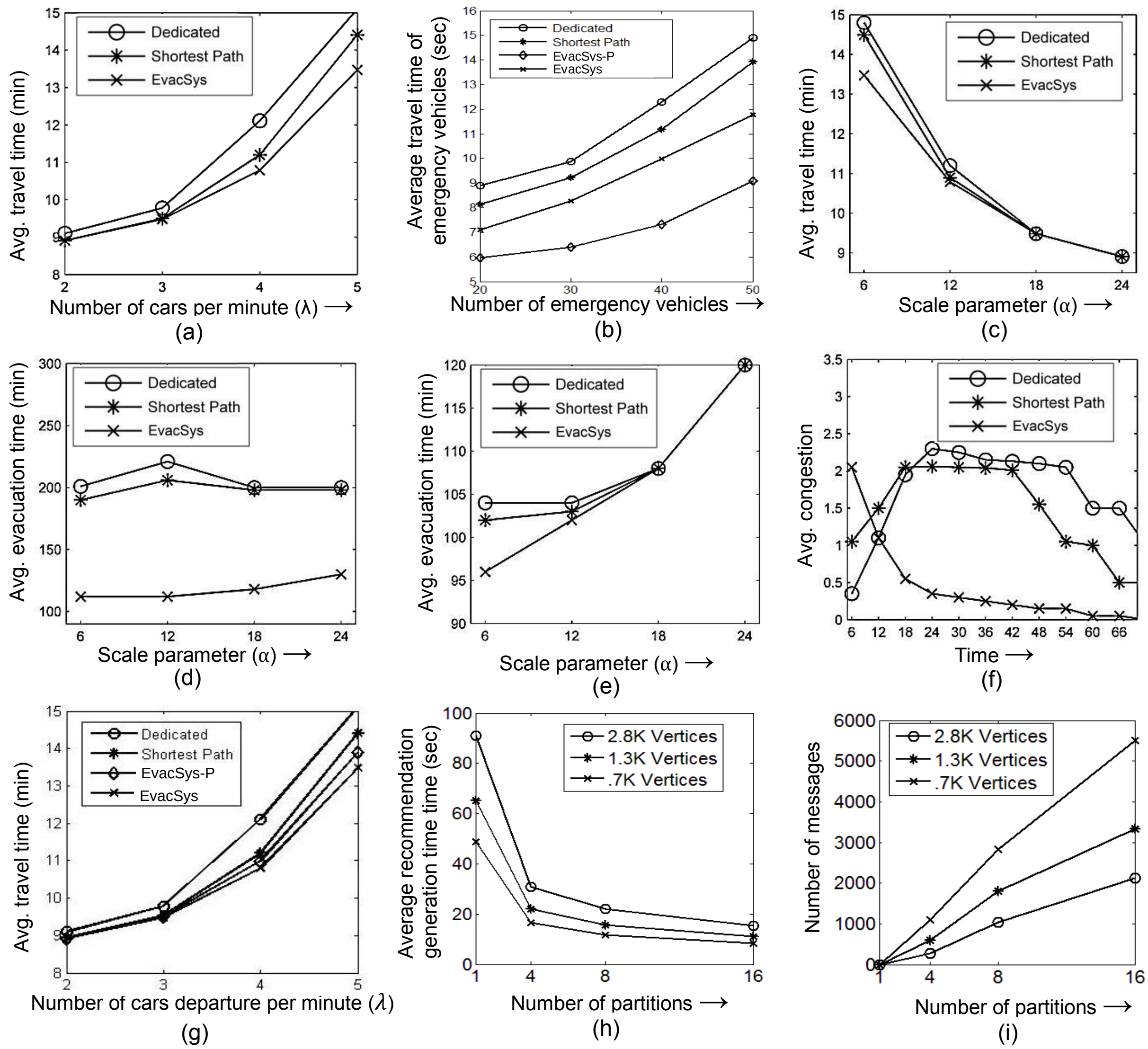


Fig. 4. Simulation results: **(a)** average travel times with varying number of departing vehicles (α=6 and β=2), **(b)** average travel times of emergency vehicles, **(c)** average travel time under normal condition with β=2 and λ=5, **(d)** average evacuations time under damaged conditions with β=2 and λ=5, **(e)** average evacuations time under normal conditions with β=2 and λ=5, **(f)** average congestion with respect to time with damaged road network α=6, β=2, and λ=5, **(g)** average travel time with increase in number of vehicles (λ), with and without prioritizing emergency vehicles, **(h)** average recommendation generation time with increase in number of partitions (processors), **(i)** number of inter-node messages exchanged with increase in number of partitions.

effect of increasing number of emergency vehicles on average travel times of emergency vehicles. Here, *EvacSys-P* approach is giving priority to emergency vehicles (as shown in Algorithm 1), whereas in simple *EvacSys*, emergency vehicles are not prioritized. It is evident that by giving priority to emergency vehicles, they take less time in travelling, compared to when emergency vehicles are not prioritized. However, both the *EvacSys-P* and *EvacSys* exhibit better performance than the *Dedicated* and *Shortest Path* due to congestion avoidance. Figs. 4(c), (d), and (e) present the effect of varying the departure time by setting values in parameter α of Weibull distribution [6]. We considered two cases, when the roads are not significantly damaged by disaster, and when the road network is damaged by disaster. As indicated in Fig. 4(c), the average travel time decreases as the departure time increases under normal road conditions. The reason for such phenomenon is that as people are taking more time in departure, there would be less congestion on the roads, and average travel times for the three approaches becomes similar. Under damaged road network (Fig. 4(d)), *EvacSys* outperforms the other two schemes because of the real-time recommendation of safe and least cost routes that is not taken into account in the other two approaches. As indicated in Fig. 4(e), under normal road conditions, when the departure time of the evacuees increases, the average evacuation times become similar for the three compared approaches. The reason is that when traffic density is low and inter-arrival times of vehicles entering road network have increased, this leads to the lesser congestion on roads. Fig. 4(f) compares the three approaches for congestion on the roads near the shelters with respect to time. In case of *Dedicated* and *Shortest-Path* approach, congestion on roads near safety shelters is less, as due to slow evacuation in such schemes, few vehicles reach the safety area initially. Alternatively, the vehicles quickly reached the roads near shelters for the *EvacSys* scheme that resulted in higher congestion in the first few minutes. As most of the vehicles have reached the shelters, the congestion decreases subsequently. Congestion level decreases relatively at lower rate in the case of *Dedicated* and *Shortest-Path*, as they do not consider traffic flow capacity while computing routes, and as a result the evacuees are guided towards short but congested roads. Fig. 4(g) indicates that when priority is given to emergency vehicles, the overall average travel time increases. This is due to the fact that evacuees' vehicles are re-routed towards alternate routes to make room for emergency vehicles, due to which average travel time of evacuees increases. Fig. 4(h) indicates the scalability performance of the proposed system. As the city map is divided into number of sub-regions (partitions) and each partition is processed in parallel on a separate node, the average recommendation time decreases. This indicates that the use of cloud-based parallel computing cluster helped in reducing the map processing time. Finally, the Fig. 4(i) shows the number of messages exchanged among cluster nodes during handovers, when vehicles leave from one region and enter the other region after crossing the boundary points.

5 CONCLUSIONS

In this magazine article, we highlighted the importance of emergency evacuation planning that is critical for saving lives during emergency scenarios. With the significant advancement in communication technologies, and the integration of modern sensors in ITS, it is now possible to extract real-time information about road and traffic conditions. However, handling huge volumes of sensory information is itself a challenging task and requires high-end computational resources. Cloud computing is an emerging platform that provides infrastructure, platform, and services necessary to solve complex problems. We presented the architecture of a cloud-based service that was meant to provide real-time route recommendations to people evacuating from a disaster site, with priority given to emergency vehicles. For empirical testing of the proposed service, we generated synthetic traffic on a real city map, and implemented our service on a cloud-based parallel computing cluster, so that the service should be able to scale on demand. The interesting findings from the results emphasized the importance of the proposed service, and how such study can benefit the disaster management bodies to plan and optimize the traffic operations during a possible evacuation. In future, we will extend our model by including more stochastic factors that are commonly observed during evacuations, such as driving under stress or fear, compliance behavior of evacuees to routing instructions, and by incorporating various types, and sizes of vehicles, building structures, to observe their effect on evacuation times.

REFERENCES

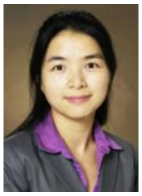
[1] http://theinstitute.ieee.org/technology-focus/technology-topic/minimizing-the-damage-from-natural-disasters, accessed on Nov, 2014.

[2] K. Chen and J. Miles, "PIARC Handbook on Intelligent Transport System," Artech House, London and Boston, 1999, ISBN 1-50852-103-2, http://road-network-operations.piarc.org/index.php?option=com_content&task=view&id=38&Itemid=71&lang

=en, assessed on Nov, 2014.
[3] L. A. Klein, M. K. Mills, and D. R.P. Gibson, "Traffic Detector Handbook: Third Edition—Volume I," FHWA Report, No. FHWA-HRT-06-108, Federal Highway Administration, McLean, VA, 2006, http://www.fhwa.dot.gov/publications /research/operations/its/06108/06108.pdf, assessed on Nov, 2014.
[4] V. W. Inman and G. W. Davis, "The Effects of In-Vehicle and Infrastructure-Based Collision Warnings at Signalized Intersections," FHWA Report, No. FHWA-HRT-09-049, Federal Highway Administration, McLean, VA, 2009, Nov, 2014.
[5] Intelligent Transportation System, http://www.its. dot. gov/, accessed on Nov, 2014.
[6] S. M. Ross. "Introduction to probability models, 9th Edition." ISBN-13: 978-0-12-598062-3, Academic Press, 2006
[7] Red River Information, http://www.ndsu.edu/fargo_geology/whyflood.htm, accessed on Nov, 2014
[8] City of Fargo website: https://www.cityoffargo.com /Maps/, accessed on Nov,, 2014
[9] North Dakota Department of Transportation (NDDOT), https://www.dot.nd.gov/, accessed on Nov, 2014
[10] Open Street Map API, http://www.openstreetmap.org/, accessed on Nov, 2014
[11] OpenMPI API, http://www.open-mpi.org/, accessed on Nov, 2014
[12] NDSU Fargo Flood Homepage, http://www.ndsu.nodak.edu/fargoflood/, accessed on Nov 2014
[13] G. Galindo and R. Batta, "Review of recent developments in OR/MS research in disaster operations management," *European Journal of Operational Research,* vol. 230, pp. 201–211, 2013.
[14] S. Kamburugamuve, L. Christiansen, and G. Fox, "A Framework for Real Time Processing of Sensor Data in the Cloud," *Journal of Sensors,* vol. 2015, no. 468047, pp. 1-11, 2015.
[15] Massimo Ficco, Roberto Pietrantuono, Stefano Russo, "Supporting ubiquitous location information in interworking 3G and wireless networks," *Communications of the ACM*, vol. 53, no. 11, pp. 116-123, 2010.

Osman Khalid completed his Ph.D. in 2014 at the North Dakota State University, Fargo. Previously, he did Masters in Computer Engineering from CASE – Pakistan. His area of research includes opportunistic networks, recommendation systems, and trust and reputation systems. He is assistant professor in COMSATS Institute of Information Technology, Abbottabad, Pakistan.

Muhammad Usman Shahid Khan completed his PhD from North Dakota State University, Fargo, USA. He received Masters in Information Security from National University of Science and Technology, NUST, Pakistan. Currently, he is research fellow at the North Dakota State University, Fargo, USA. His areas of interest are recommendation systems, data mining, cognitive radio networks, and network security



Ying Huang received her Ph. D. in civil engineering from Missouri University of Science and Technology, Rolla, USA, in 2012 and M.S. from Harbin Institute of technology, China in 2008. She is currently an assistant professor at the Department of Civil and Environmental Engineering, North Dakota State University. She has more than 30 publications. Her areas of interest include structural health monitoring/smart structures for transportation infrastructure, intelligent transportation systems, applications of adaptive and smart materials, and multi-hazard assessment and mitigation, etc.

Samee U. Khan is an associate professor of electrical and computer engineering at North Dakota State University. His research interests include optimization, robustness, and security of cloud, grid, cluster, and big data computing, social networks, wired and wireless networks, power systems, smart grids, and optical networks. Khan has a PhD in computer science from the University of Texas, Arlington. He's a senior member of IEEE. Contact him at samee.khan@ndsu.edu.

Albert Y. Zomaya is the chair professor of highperformance computing and networking in the School of Information Technologies at Sydney University. His research interests are in the areas of complex systems, parallel and distributed computing, and green computing. Zomaya is editor in chief of IEEE Transactions on Computers and is the recipient of the IEEE TCPP Outstanding Service Award and the IEEE TCSC Medal for Excellence in Scalable Computing. He has a PhD in control engineering from Sheffield University, UK. Zomaya is a Fellow of AAAS, IEEE, and IET (UK). Contact him at zomaya@it.usyd.edu.au