Probability in the Engineering and Informational Sciences, **32**, 2018, 340–352.
doi:10.1017/S0269964816000115

ENERGY-AWARE NAVIGATION IN LARGE-SCALE EVACUATION USING G-NETWORKS

Huibo Bi and Omer H. Abdelrahman
Intelligent Systems and Networks Group, Dept. of Electrical and Electronic Engineering, Imperial College, London SW7 2AZ, UK
E-mail: {huibo.bi12,o.abd06}@imperial.ac.uk

Previous studies on emergency management of large-scale urban networks have commonly concentrated on system development to off-load intensive computations to remote cloud servers or improving communication quality during a disaster and ignored the effect of energy consumption of vehicles, which can play a vital role in large-scale evacuation owing to the disruptions in energy supply. Hence, in this paper we propose a cloud-enabled navigation system to direct vehicles to safe areas in the aftermath of a disaster in an energy and time efficient fashion. A G-network model is employed to mimic the behaviors and interactions between individual vehicles and the navigation system, and analyze the effect of re-routing decisions toward the vehicles. A gradient descent optimization algorithm is used to gradually reduce the evacuation time and fuel consumption of vehicles by optimizing the probabilistic choices of linked road segments at each intersection. The re-routing decisions arrive at the intersections periodically and will expire after a short period. When a vehicle reaches an intersection, if the latest re-routing decision has not expired, the vehicle will follow this advice, otherwise, the vehicle will stick to the shortest path to its destination. The experimental results indicate that the proposed algorithm can reduce the evacuation time and the overall fuel utilization especially when the number of evacuated vehicles is large.

1. INTRODUCTION

Compared with emergency evacuations in built environment which primarily demand time-critical responses, large-scale evacuations trigger sudden massive demands on the transport network and require efficient traffic management and resource allocation mechanisms. During urban-scale evacuations, traffic infrastructures including roads built for daily activities can suffer high congestion levels which prolong the evacuation process. In addition, non-eco-driving at less fuel-efficient speeds as well as the frequent acceleration and deceleration caused by braking can consume a large amount of fuel, which has a significant impact on large-scale evacuations due to the disruption in energy supply. Hence, in this paper we propose a cloud-enabled system to guide vehicles to safe areas in the aftermath of large-scale disasters with the aid of a queueing network based model, namely G-network [9,10]. The G-network model is used to comprehensively mimic the behaviors and interactions of individual vehicles and the navigation system in an evacuation process and analyze the effect of the re-routing decisions toward the vehicles. A gradient descent algorithm is employed to reduce the evacuation time and fuel consumption of vehicles by optimizing the probabilistic choices of linked road segments at each intersection.

The remainder of the paper is organized as follows. We first review related work in Section 2, then describe the system framework in Section 3.1 and the approximation models in Sections 3.2 and 3.3. The energy efficiency and latency routing metrics are introduced in Section 4. Using the gradient descent optimization in Section 5, the long-term probabilistic routing decisions for vehicles are selected so as to achieve energy and time efficiency. The simulation model and assumptions are discussed in Section 6, and the related and experimental results are presented in Section 7. Finally, we draw conclusions in Section 8.

2. RELATED WORK

2.1. Emergency Navigation in Large-scale Disasters

Previous work on large-scale evacuations has mainly focused on system development to off-load intensive computations to the remote Cloud or maintain and improve communication quality during a disaster. For instance, early warning systems such as [27,31] leverage existing public cloud services to gather and disperse multimedia emergency information among evacuees; users can upload observed emergency situations via portable devices and share within the community. Emergency navigation systems such as [17] employ camera-equipped smart phones to take and upload snapshots to cloud servers for localization. Based on the distribution of evacuees, a Cognitive Packet Network [14,20,21] algorithm is used to search for safe evacuation routes with shortest times to exits. Vehicular Ad hoc Networks (VANETs) and cloud computing technologies are utilized in [2] to establish an urban-level intelligent disaster management system; information from multiply sources and locations is gathered to make effective decisions, and the Lighthill–Whitham–Richards model is used to simulate the traffic flows in the city. On the other hand, emergency communication has also raised considerable interests since it plays a critical role in emergency response. For example, [26] utilizes opportunistic communications (Oppcomms) among portable devices carried by evacuees to disperse emergency messages in a “store-carry-forward” manner, while [30] employs application-layer multicast to rapidly deliver emergency traffic without the support of a dedicated network infrastructure. However, in large-scale disasters such as earthquakes, energy utilization can also have a significant influence on the evacuation process due to the disruptions in energy supply. Indeed, fuel shortage has been reported in several recent earthquake-related disasters [5], but this aspect has been largely ignored in the emergency management literature.

2.2. Energy-Efficient Traffic Management

Energy-efficient traffic management has received considerable attention recently, due to strong regulations and ambitious targets concerning the reduction of green house gas emissions. In particular, a number of route planning algorithms have been proposed to reduce energy consumption of vehicles during daily usage, by integrating various macroscopic and microscopic scale traffic/energy models [4]. The work in [39] employs Dijkstra’s algorithm to calculate fuel-efficient paths in terms of the physical length and the estimated fuel consumption on each road segment which is estimated from the average travelling speed. Similarly, an energy-efficient routing algorithm for electric cars is presented in [32], where a cost function is defined for each edge in the network in terms of road slope, vehicle speed, vehicle acceleration, journey time and road surface conditions; since edge weights may evaluate negative

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

values which cannot be handled by Dijkstra's or A* algorithms, the Bellman-Ford-algorithm is employed to compute the most energy-efficient paths. A fuel-efficient driving strategy for high-way vehicles is proposed in [29], using the Max–Min Ant system [36] to optimize the vehicle speed and acceleration commands. By re-routing traffic and giving speed advice at signalized intersections, [37] presents a comprehensive intelligent transportation system to balance traffic and reduce fuel consumption with the aid of vehicle-to-vehicle and vehicle-to-infrastructure communication technologies; experimental results show that the proposed system can significantly reduce the number of stops and the average waiting time at intersections as well as the overall fuel consumption. In summary, although much work has been dedicated to reducing the energy consumption of vehicles for daily usage, to the best of our knowledge, there are no studies that consider optimizing both energy and latency in large-scale evacuations.

3. THE NAVIGATION SYSTEM

3.1. System Architecture

Figure 1 shows the architecture of our proposed system. We assume that a sensor node (SN) and a communication node (CN) are pre-installed at each intersection. SNs are roadside cameras that can detect and count vehicles in roadways while CNs are communication units that are responsible for information exchanges between vehicles and cloud servers. When the evacuation process starts, if the existing communication infrastructure is non-operational, we assume a few cloud access points can be quickly deployed to maintain communication between the cloud servers and the CNs. The collected sensory data is periodically uploaded to the cloud servers, which are responsible for calculating the shortest path from each intersection to the entrance of the safe area, as well as running the G-network based algorithm to compute possible re-routing decisions at each intersection. A re-routing decision activates a timeout upon arriving at a CN: when a vehicle reaches an intersection and the latest re-routing decision has not expired, it will follow the re-routing advice calculated by the G-network based algorithm; otherwise, it will follow the shortest path to the safe area. Thus, we call our approach the G-network assisted Dijkstra's shortest path (GDSP) algorithm. We assume evacuees can use portable devices or vehicle-mounted devices to communicate with the CNs and obtain support at intersections during the evacuation.

3.2. System Approximation Model

The transportation network of our targeted area is represented by a directed graph consisting of nodes and edges. Nodes are Points of Interest (PoIs) such as intersections where vehicles may congregate, and are modeled as a first-in-first-out (FIFO) queue. Edges are the physical links between PoIs, and are modeled as a processor sharing system [7] to represent the facts that the time taken to traverse an edge is affected by the number of vehicles on it, and that the order in which vehicles reach the next PoI is not necessarily FIFO. Rather than choosing the shortest paths for vehicles to reach the safe areas, we use a G-network model to optimize in a centralized manner the probabilistic choices among all possible roads to achieve both time and energy efficiency.

G-networks [11], which were inspired by the random neural network [8], are a class of queueing network models with additional control capabilities such as negative customers [9] that remove normal positive customers from the system, batch removals [10], triggers [10] and resets [19]. G-networks have been used in a wide range of applications, including describing the workload in computer systems [24,33], realizing energy efficiency in packet

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

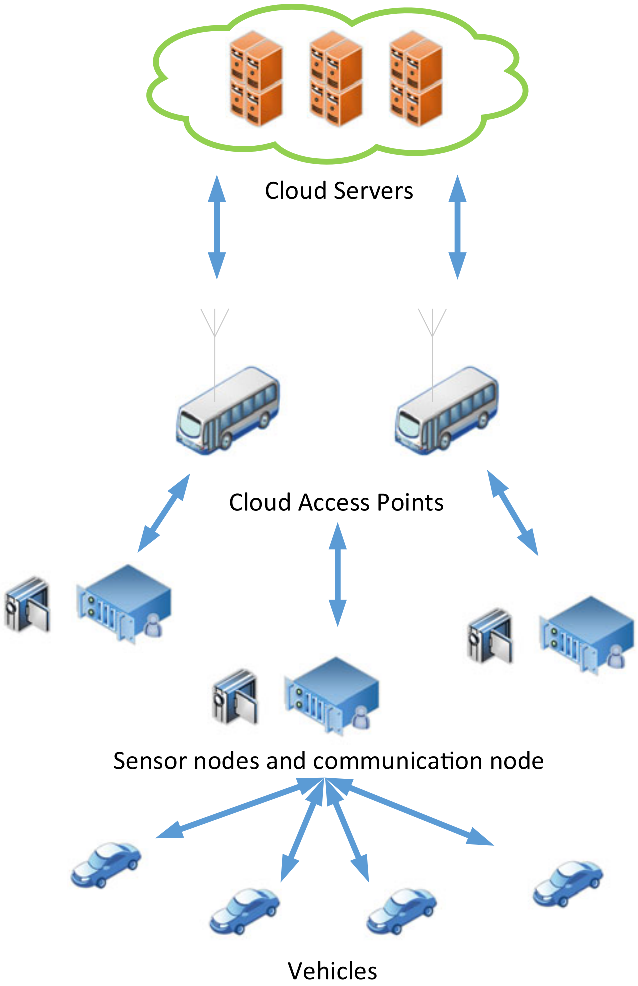

FIGURE 1. The architecture of the proposed framework.

networks [23], as well as modeling energy systems [16], populations of biological agents [12] and gene regulatory networks [13].

The model we use to capture the dynamics of an evacuation process is based on [10], and has only positive customers and triggers representing vehicles and re-routing decisions, respectively. We assume that positive customers (vehicles) that have just started to move will join their nearest PoIs, and this "external" arrival of vehicles to n_i occurs at an average rate of Λ_{n_i}. The average service rate of vehicles at PoI n_i is denoted by r_{n_i} which depends on the physical characteristics of the node including the size and number of intersecting roads. A vehicle which is leaving PoI n_i will either head to another connected PoI n_j with probability $P(n_i, n_j)$ or pull up at the side of the road with probability d_{n_i}, where $d_{n_i} + \sum_{j=1}^{N} [P(n_i, n_j)] = 1$ and N is the total number of PoIs.

In addition to the above navigation decisions which are taken by the evacuees, re-routing decisions from the cloud-based system arrive to PoI n_i with average rate $\lambda^-_{n_i}$, instructing the leading vehicle to move to PoI n_j with probability $Q(n_i, n_j)$, where $\sum_{j=1}^{N} Q(n_i, n_j) = 1$. This probability is a key parameter to be optimized in our system, as previous research [23]

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

has indicated that it can significantly affect performance. With these assumptions, the steady-state probability that a PoI n_i has one or more vehicles is given by [10]:

$$q_{n_i} = \frac{\lambda^+_{n_i}}{r_{n_i} + \lambda^-_{n_i}}, \tag{1}$$

where $\lambda^+_{n_i}$ is the total average arrival rate of vehicles to PoI n_i, including vehicles that were previously parked or at other PoIs:

$$\lambda^+_{n_i} = \Lambda_{n_i} + \sum_{j=1}^{N} q_{n_j}[r_{n_j}P(n_j, n_i) + \lambda^-_{n_j}Q(n_j, n_i)]. \tag{2}$$

Notice that the quantities q_{n_i} are coupled, and therefore (1) is a non-linear equation that can be solved numerically.

For the edges of the network, the arrival rate of vehicles to a road segment e_{ij} connecting PoIs n_i and n_j can be calculated as:

$$\lambda^+_{e_{ij}} = q_{n_i}[r_{n_i}P(n_i, n_j) + \lambda^-_{n_i}Q(n_i, n_j)]. \tag{3}$$

Furthermore, the service rate of the road segment e_{ij} is approximated by:

$$r_{e_{ij}} = \frac{V_{e_{ij}}}{L_{e_{ij}}}, \tag{4}$$

where $L_{e_{ij}}$ is the physical length, and $V_{e_{ij}}$ is the average speed of a vehicle when no other vehicles are concurrently using the road segment. The utilization of the road segment then becomes:

$$q_{e_{ij}} = \frac{\lambda^+_{e_{ij}}}{r_{e_{ij}}} = q_{n_i}\frac{[r_{n_i}P(n_i, n_j) + \lambda^-_{n_i}Q(n_i, n_j)]L_{e_{ij}}}{V_{e_{ij}}} \equiv q_{n_i}R(n_i, n_j). \tag{5}$$

Let $K_{n_i}(t)$ and $K_{e_{ij}}(t)$ denote the number of vehicles at time t in PoI n_i and edge e_{ij}, respectively, then the joint equilibrium distribution of the number of vehicles in the network has a product form [10] and is given by:

$$\Pr(K_{n_i} = k_i,\ K_{e_{ij}} = k_{ij},\ i, j = 1, 2, \ldots, N) = \prod_{i=1}^{N} q^{k_i}_{n_i}(1 - q_{n_i}) \prod_{i,j=1}^{N} q^{k_{ij}}_{e_{ij}}(1 - q_{e_{ij}}). \tag{6}$$

From the above expression, one can compute interesting measures such as the average number of vehicles in each intersection and road segment, the response time (or equivalently the travel time) along a road segment, and the average speed of vehicles along a road segment which is obtained as the ratio of the physical length $L_{e_{ij}}$ and average response time.

3.3. Fuel Consumption Models

The fuel consumption of a vehicle is well known to be significantly affected by the speed, acceleration level [39] and loading weight [38]. In this section, we employ two separate models to characterize fuel consumption on the road segments and intersections. These models are used by both the re-routing algorithm in estimating the energy cost of a decision as well as the simulator in calculating the total cost of the evacuation process. However, the algorithm relies on the G-network model in order to estimate the speed of a vehicle which is then used

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

to compute fuel consumption, while the simulator employs a more realistic vehicle mobility model as described in Section 6.1.

The work in [39] establishes mesoscopic models for light-duty gasoline vehicles, mid-duty diesel vehicles and heavy-duty diesel vehicles based on collected emission data. The models reflect the relation among average travel speed, travel distance and fuel consumption of a vehicle, and are suitable for modeling consumption on road segments. For example, the fuel consumption factor FC_L in kg/100 km for light-duty gasoline vehicles is described in terms of the average travel speed $\bar{v}$ km/hr as follows:

$$\mathrm{FC}_L = \frac{125.015}{\bar{v}} - 0.097\bar{v} + 9.22 * 10^{-4}\bar{v}^2 + 7.056. \tag{7}$$

On the other hand, fuel consumption at the intersections is mainly affected by braking events, which occur for example, when a vehicle reaches an intersection with a non-zero queue. Although the deceleration part of the braking cycle does not have an obvious impact on fuel consumption, remarkable fuel utilization is observed during acceleration when the vehicle operator releases the brake pedal and starts to depress the accelerator. Based on [3,28], the vehicle specific power (VSP), defined as the instantaneous power per unit mass of a vehicle and measured in kW/Metric Ton, for light-duty or mid-duty vehicles is given by:

$$VSP_{lm} = 0.037v + 2.358 * 10^{-5}av + 6.859 * 10^{-6}v^3, \tag{8}$$

where v is the instantaneous speed of the vehicle in km/h and a is the acceleration in km/hr^2. The energy consumption of the vehicle is then obtained by multiplying VSP_{lm} by the weight of the vehicle, and integrating over the duration of the acceleration period. The total fuel consumed in this process is subsequently calculated from the energy density of the fuel.

4. ROUTING METRICS

The QoS requirements during a large-scale evacuation are typically related to evacuation time which, however, can be influenced by the availability of energy supply. Hence, in this section we define a compound objective function that includes both evacuation time D_t and fuel consumption C_t. Specifically, to achieve energy efficiency while maintaining an acceptable level of delay, we combine these two metrics as follows:

$$F_t = C_t + \epsilon D_t, \tag{9}$$

where ϵ is a constant that determines the relative importance of the two metrics. The above objective function is computed based on the G-network representation of the network, and is optimized periodically during the evacuation process using a gradient descent algorithm.

4.1. Time-Aware Metric

The total evacuation time in the network includes delays at both intersections and road segments, and depends on the congestion level. The average number of vehicles at an intersection n_i or a road segment e_{ij} can be derived directly from (6) yielding:

$$N_{n_i} = \frac{q_{n_i}}{1 - q_{n_i}}, \quad N_{e_{ij}} = \frac{q_{e_{ij}}}{1 - q_{e_{ij}}} \tag{10}$$

Using Little's formula, the average traversal times are given by:

$$D_{n_i} = \frac{N_{n_i}}{\lambda^+_{n_i}}, \quad D_{e_{ij}} = \frac{N_{e_{ij}}}{\lambda^+_{e_{ij}}} \tag{11}$$

while the total average delay experienced by a vehicle in the network is:

$$D_t = \frac{\sum_{i=1}^{N} N_{n_i} + \sum_{i,j=1}^{N} N_{e_{ij}}}{\sum_{i=1}^{N} \Lambda_{n_i}} \qquad (12)$$

where the numerator is the total average number of vehicles in the network, and denominator is the total rate at which vehicles join the network (e.g., start the evacuation or arrive from outside the evacuation area).

4.2. Energy-Aware Metric

Since fuel consumption of vehicles depends not only on travelled distance but also on speed and acceleration, the energy-aware metric should take into account congestion at both road segments and intersections. Using (7), the total average fuel consumption, measured in kg, by the vehicles traversing the road segment e_{ij} is:

$$C_{e_{ij}} = N_{e_{ij}} \frac{L_{e_{ij}}}{100} \mathrm{FC}_L \qquad (13)$$

where FC_L is a function of the average speed of vehicles $v_{e_{ij}}$ that can be estimated from the queueing model as:

$$v_{e_{ij}} = \frac{L_{e_{ij}}}{D_{e_{ij}}} = \frac{L_{e_{ij}} \lambda^{+}_{e_{ij}}}{N_{e_{ij}}} \qquad (14)$$

Next we describe how the G-network based algorithm estimates fuel consumption at the intersections. Due to the stop-and-go behavior in traffic waves at each intersection, we can relate on average the number of breaking events to the number of vehicles $N_{e_{ij}}$. To further simplify computations and the data collected by the algorithm, it is assumed that whenever a vehicle encounters congestion (at least one other vehicle at the intersection), then it will decelerate and accelerate once. Thus, if we denote by c_{lm} the fuel consumed by a vehicle in this process, which will be estimated in Section 6.2 using the energy model (8) and the mobility model of the simulator (26), then the total average fuel consumption at intersection n_i becomes:

$$C_{n_i} = \sum_{k=1}^{\infty} (k-1) c_{lm} \Pr(K_i = k) = \sum_{k=1}^{\infty} (k-1) c_{lm} (1 - q_{n_i}) q_{n_i}^{k} = c_{lm} \frac{q_{n_i}^{2}}{1 - q_{n_i}}$$

and the total fuel consumption in the network is given by:

$$C_t = \sum_{i=1}^{N} C_{n_i} + \sum_{i,j=1}^{N} C_{e_{ij}}. \qquad (15)$$

5. GRADIENT DESCENT OPTIMIZATION

To minimize the objective function (9), we use a gradient-based algorithm to select appropriate $Q(n_i, n_j)$ which is the probabilistic re-routing decision from intersection n_i to

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

intersection n_j, where $i, j = 1, 2, \ldots, N$. Using the above analysis, (9) can be expressed as:

$$F_t = \sum_{i,j=1}^{N} \left[1.25015 \frac{N_{e_{ij}}^2}{\lambda_{e_{ij}}^+} - 9.7 * 10^{-4} L_{e_{ij}}^2 \lambda_{e_{ij}}^+ + 9.22 * 10^{-6} \frac{L_{e_{ij}}^3 \lambda_{e_{ij}}^{+\ 2}}{N_{e_{ij}}} + 7.056 * 10^{-2} L_{e_{ij}} N_{e_{ij}} \right] \\ + \sum_{i=1}^{N} c_{lm} \frac{q_{n_i}^2}{1 - q_{n_i}} + \epsilon \left(\sum_{i=1}^{N} N_{n_i} + \sum_{i,j=1}^{N} N_{e_{ij}} \right), \qquad \textbf{(16)}$$

where the denominator of D_t is incorporated into ϵ since it does not depend on the optimization parameters. To evaluate the partial derivative of F_t with respect to $Q(n_x, n_y)$ – which is difficult to compute directly—we utilize the chain rule:

$$\frac{\partial F_t}{\partial Q(n_x, n_y)} = \sum_{k=1}^{N} \frac{\partial F_t}{\partial q_{n_k}} \frac{\partial q_{n_k}}{\partial Q(n_x, n_y)} \qquad \textbf{(17)}$$

Using (2)–(5) and (10) we can write (16) as follows:

$$F_t = \sum_{i,j=1}^{N} \left\{ \frac{1.25015 L_{e_{ij}}}{V_{e_{ij}} [1 - q_{n_i} R(n_i, n_j)]^2} - 9.7 * 10^{-4} L_{e_{ij}} V_{e_{ij}} \right. \\ \left. + 9.220 * 10^{-6} L_{e_{ij}} V_{e_{ij}}^2 [1 - q_{n_i} R(n_i, n_j)] + \frac{7.056 * 10^{-2} L_{e_{ij}} + \epsilon}{1 - q_{n_i} R(n_i, n_j)} \right\} q_{n_i} R(n_i, n_j) \\ + \sum_{i=1}^{N} \left[c_{lm} \frac{q_{n_i}^2}{1 - q_{n_i}} + \epsilon \frac{q_{n_i}}{1 - q_{n_i}} \right]. \qquad \textbf{(18)}$$

Taking the partial derivative of the above expression with respect to a specific q_{n_k} yields:

$$\frac{\partial F_t}{\partial q_{n_k}} = \sum_{j=1}^{N} R(n_k, n_j) \left\{ \frac{1.25015 L_{e_{kj}} [1 + q_{n_k} R(n_k, n_j)]}{V_{e_{kj}} [1 - q_{n_k} R(n_k, n_j)]^3} - 9.7 * 10^{-4} L_{e_{kj}} V_{e_{kj}} \right. \\ + 9.220 * 10^{-6} L_{e_{kj}} V_{e_{ij}}^2 [1 - 2 q_{n_k} R(n_k, n_j)] \\ \left. + \frac{7.056 * 10^{-2} L_{e_{kj}} + \epsilon}{[1 - q_{n_k} R(n_k, n_j)]^2} \right\} + \frac{c_{lm} q_{n_k} (2 - q_{n_k}) + \epsilon}{(1 - q_{n_k})^2}. \qquad \textbf{(19)}$$

Also, the partial derivative of q_{n_k} for a specific $Q(n_x, n_y)$ is given by:

$$\frac{\partial q_{n_k}}{\partial Q(n_x, n_y)} = \frac{1}{r_{n_k} + \lambda_{n_k}^-} \sum_{j=1}^{N} \left\{ \lambda_{n_j}^- q_{n_j} \frac{\partial Q(n_j, n_k)}{\partial Q(n_x, n_y)} \right. \\ \left. + \left[r_{n_j} P(n_j, n_k) + \lambda_{n_j}^- Q(n_j, n_k) \right] \frac{\partial q_{n_j}}{\partial Q(n_x, n_y)} \right\}. \qquad \textbf{(20)}$$

Notice that:

$$\frac{\partial Q(n_j, n_k)}{\partial Q(n_x, n_y)} = \begin{cases} 1, & \text{if } j = x,\ k = y, \\ -1, & \text{if } j = x,\ k \neq y, \\ 0, & \text{otherwise}, \end{cases} \qquad \textbf{(21)}$$

where the second case arises from the identity $Q_{n_x, n_y} = 1 - \sum_{k \neq y} Q_{n_x, n_k}$. Now define the $N \times N$ matrices $\mathbf{A}(i, j) = r_{n_j} P(n_j, n_i)$, $\mathbf{B}(i, j) = \lambda_{n_j}^- Q(n_j, n_i)$, and

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

$\mathbf{C} = \mathrm{diag}(r_{n_1} + \lambda^-_{n_1}, \ldots, r_{n_N} + \lambda^-_{n_N})^{-1}$, then we can write (20) in a more compact form as:

$$\frac{\partial \mathbf{q_n}}{\partial Q(n_x, n_y)} = \mathbf{C}(\mathbf{A^T} + \mathbf{B^T}) \frac{\partial \mathbf{q_n}}{\partial Q(n_x, n_y)} + \mathbf{C}\mathbf{H^T_{xy}}\mathbf{q_n}, \tag{22}$$

where $\mathbf{H_{xy}}$ is a $N \times N$ matrix whose elements are:

$$\mathbf{H_{xy}}(i,j) = \begin{cases} \lambda^-_{n_i}, & i = x, j = y, \\ -\lambda^-_{n_i}, & i = x, j \neq y, \\ 0, & \text{otherwise}. \end{cases} \tag{23}$$

Combining (19) and (22), the partial derivative of the cost function becomes:

$$\frac{\partial F_t}{\partial Q(n_x, n_y)} = \left(\frac{\partial F_t}{\partial \mathbf{q_n}}\right)^T \left[\mathbf{I_N} - \mathbf{C}(\mathbf{A^T} + \mathbf{B^T})\right]^{-1} \mathbf{C}\mathbf{H^T_{xy}}\mathbf{q_n}, \tag{24}$$

where $\mathbf{I_N}$ is the $N \times N$ identity matrix. Finally, we compute $Q(n_x, n_y)$ using the following iterations:

$$Q_{n+1}(n_x, n_y) = Q_n(n_x, n_y) - \eta \frac{\partial F_t}{\partial Q_n(n_x, n_y)}, \tag{25}$$

where $\eta > 0$ is the learning rate.

6. SIMULATION MODEL AND ASSUMPTIONS

To evaluate the proposed routing scheme for vehicles, we employ a multi-agent based simulation tool, the distributed building evacuation simulator (DBES) [6] to conduct earthquake-related simulations. The area under consideration is located in London as shown in Figure 2. Vertices represent PoIs, which are broad areas where vehicles can queue in lines and receive suggestions, while edges represent available roads in the aftermath of a disaster. As mentioned previously, each intersection is equipped with a SN which collects on-site information and a CN which uploads the gathered information and transmits advice to nearby vehicles. The CNs can communicate with each other and a few cloud access points, and can be quickly deployed in the affected area in case of infrastructure failure. Hence, certain cloud servers outside of the disaster area can be available to provide optimal solutions in a centralized manner.

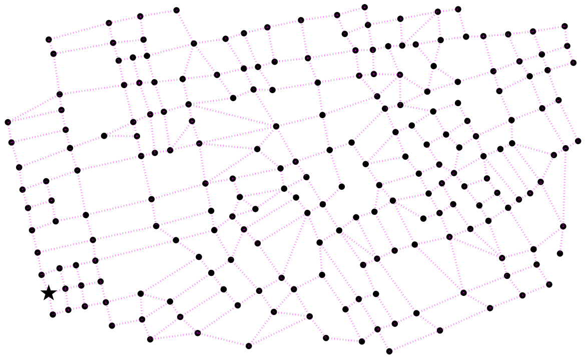

FIGURE 2. Transportation network of the targeted area in London city.

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

The re-routing decisions calculated by the cloud servers arrive at the CNs every 4 seconds (i.e., $\lambda^-_{n_i} = 1/4$) and the time-out of the re-routing decisions is set to 1 s. Initially, the simulated vehicles are randomly distributed in the area. Moreover, all the simulated vehicles are assumed to be light-duty with identical physical attributes, and the gross vehicle weight rating is 3 metric tons.

6.1. Vehicle Mobility

According to [34,35], the average travel speed of a vehicle highly depends on the density of the road segment it is on. Thus, we use the mobility model proposed in [35] to represent the travel speed of vehicles in our simulator. Specifically, if a vehicle arrives at a road segment without other vehicles, we assume that the vehicle will move at the most fuel-efficient speed; otherwise, the speed of the vehicle will depend linearly on congestion as follows:

$$v = \begin{cases} v_o, & \text{if no other vehicle is present,} \\ \sigma \frac{v_o}{c}(c+1-n), & \text{otherwise,} \end{cases} \tag{26}$$

where σ is a constant, v_o is the most fuel-efficient speed of the vehicle, n is the total number of vehicles, and c is the capacity of the road segment which depends on its maximum density, number of lanes, and length.

On the other hand, drivers at intersections are affected by the space headway, and will perform either the free-flow behavior or the car-following behavior [1]. In particular, if an intersection is unoccupied, the driver can choose its desired speed, while if the intersection is in use by at least one vehicle, then the speed and acceleration are governed by the leader [1]. We assume that the leader and followed vehicles at an intersection will first accelerate to 60 km/hr after a braking event, at an acceleration rate of 64,800 km/h^2 (18 km/hr/s).

6.2. Fuel Consumption

The fuel consumption of vehicles travelling on a road segment is calculated in the simulator based on the mobility model in (26) and the fuel consumption factor FC$_L$ in (7). Also, since the gross vehicle weight rating is assumed to be 3 metric ton, the power consumption of vehicles at intersections (in kW) can be computed from (8) as:

$$\begin{aligned} P_{lm} &= VSP_{lm} \times \text{ weight} \\ &= 0.111v + (7.074 \times 10^{-5})av + (2.058 \times 10^{-5})v^3. \end{aligned} \tag{27}$$

The time for a stopped vehicle to accelerate to 60 km/h at a rate of 18 km/hr/s is $9.26 * 10^{-4}$ hr, and the energy cost of this process in KWh can be calculated by integrating the above expression and noting that the speed of the vehicle v is related to acceleration a and time t by $v = at$ (starts from a standstill):

$$\begin{aligned} E_{lm} = \int_0^{9.26\times10^{-4}} & P_{lm}dt = 0.111at + 7.074 \times 10^{-5}a^2t \\ & + 2.058 \times 10^{-5}a^3t^3\, dt = 0.1314\, kWh \end{aligned} \tag{28}$$

Since energy density of gasoline is about 44.4 MJ/kg or 12.432 kWh/kg, the fuel consumption for a single braking event is:

$$c_{lm} = E_{lm} \times \text{energy density} \approx 0.01\,\text{kg} \tag{29}$$

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

7. RESULTS AND DISCUSSION

The experiments are conducted for scenarios with 50, 100 and 500 vehicles representing light, moderate and heavy traffic conditions, respectively. We compare the performance of our G-network assisted GDSP approach against the standard DSP algorithm.

Figure 3 shows the average evacuation time per vehicle for both GDSP and DSP. The simulation results indicate that GDSP achieves similar performance to DSP at low traffic density (50 vehicles) and shorter evacuation times at higher traffic densities. This is because DSP tends to guide all the vehicles toward the shortest path, causing severe congestion. As a result, the travel speed of vehicles is reduced and evacuation process is prolonged. On the other hand, GSDP reduces the likelihood of such bottlenecks by distributing vehicles at each intersection.

The average overall fuel consumption during the evacuation process is shown in Figure 4, where GDSP achieves less fuel consumption in all of the three scenarios. The reason is that, when using GDSP, the probabilistic choices to the linked road segment are optimized

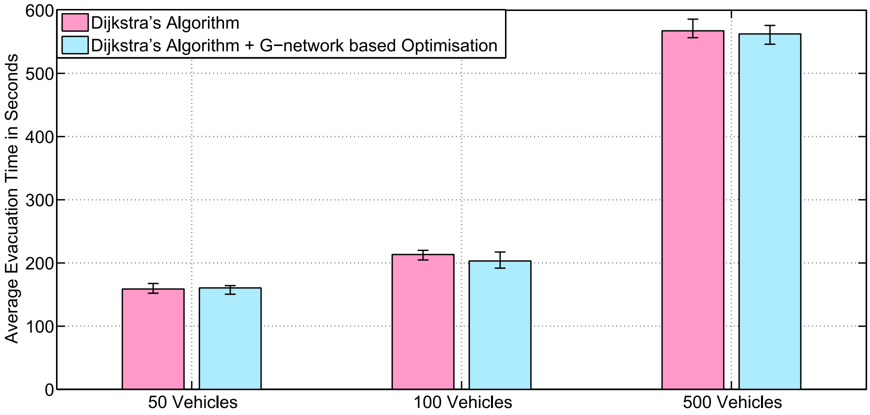

FIGURE 3. The average evacuation time of different scenarios in seconds. The results are the average of 5 randomized simulation runs, and error bars shows the min/max result in any of the five simulation runs.

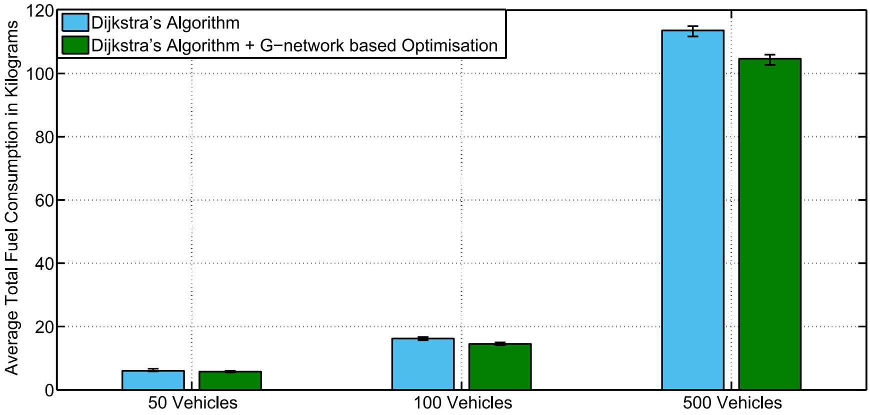

FIGURE 4. The average total fuel utilization of different scenarios in kilograms. The results are the average of 5 randomized simulation runs, and error bars shows the min/max result in any of the five simulation runs.

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press

periodically at the intersections, allowing vehicles to be re-routed to unoccupied paths where they can travel at more fuel-efficient speeds. In contrast, DSP guides vehicles to the congested shortest paths, where vehicles perform braking operations frequently, thus consuming more fuel.

8. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a cloud based architecture and an algorithm to direct vehicles to safe areas in the aftermath of a large-scale disaster. A G-network model is used by the algorithm to represent the movement of vehicles and their interactions with the navigation system. To reduce the overall fuel consumption and evacuation time, re-routing decisions at intersections are computed based on a compound routing metric, and gradually optimized using a gradient descent algorithm. Simulation results indicate that our approach achieves better fuel efficiency and evacuation time than when vehicles follow shortest paths to destinations, and the improvements in performance become more pronounced as congestion increases. In future work, we will consider the use of heuristic routing algorithms such as the Cognitive Packet Network [15,22,25] to guide vehicles, which eliminates the need to run our solution on top of a shortest path routing algorithm which can be time consuming and computationally complex in a large-scale network. Moreover, we believe that the proposed G-network model can be an effective tool in optimizing the daily traffic flows in urban networks. Indeed, current studies on road traffic optimization typically make a simplifying assumption of a fixed number of origin and destination pairs. On the other hand, G-networks can achieve long-term objectives while allowing randomness in both the destinations of vehicles and the individual's choice at each interaction [18].

References

1. Ahmed, K.I. (1999). Modeling drivers' acceleration and lane changing behavior. Ph.D. dissertation, Massachusetts Institute of Technology.
2. Alazawi, Z., Altowaijri, S., Mehmood, R., & Abdljabar, M.B. (2011). Intelligent disaster management system based on cloud-enabled vehicular networks. In *ITS Telecommunications (ITST), 2011 11th International Conference on*. IEEE, pp. 361–368.
3. Andrei, P. (2001). Real world heavy-duty vehicle emissions modeling. Ph.D. dissertation, West Virginia University.
4. Barceló, J. et al. (2010). *Fundamentals of traffic simulation*. New York: Springer, vol. 145.
5. Brown, R. (2015). Nepal facing critical fuel shortage due to blocked supply routes, fears of another humanitarian disaster. ABC news report, October 2015. [Online]. Available: http://www.abc.net.au/news/2015-10-22/nepal-faces-critical-fuel-shortage-due-to-blocked-supply-routes/6874870
6. Dimakis, N., Filippoupolitis, A., & Gelenbe, E. (2010). Distributed building evacuation simulator for smart emergency management. *Computer Journal* 53(9): 1384–1400. [Online]. Available: http://dx.doi.org/10.1093/comjnl/bxq012
7. Fourneau, J.-M., Gelenbe, E., & Suros, R. (1996). G-networks with multiple classes of negative and positive customers. *Theoretical Computer Science* 155(1): 141–156.
8. Gelenbe, E. (1989). Random neural networks with negative and positive signals and product form solution. *Neural Computation* 1(4): 502–510.
9. Gelenbe, E. (1991). Product-form queueing networks with negative and positive customers. *Journal of Applied Probability* 28(3): 656–663.
10. Gelenbe, E. (1993). G-networks with triggered customer movement. *Journal of Applied Probability* 30(3): 742–748.
11. Gelenbe, E. (2000). The first decade of g-networks. *European Journal of Operational Research* 126(2): 231–232. [Online]. Available: http://dx.doi.org/10.1016/S0377-2217(99)00475-0
12. Gelenbe, E. (2007). Dealing with software viruses: a biological paradigm. *Information Security Technical Report* 12(4): 242–250.
13. Gelenbe, E. (2007). Steady-state solution of probabilistic gene regulatory networks. *Physical Review E* 76(3): 31903.

14. Gelenbe, E. (2009). Steps toward self-aware networks. *Communications of the ACM* 52(7): 66–75.
15. Gelenbe, E. (2011). Self-aware networks. In *2011 Fifth IEEE International Conference on Self-Adaptive and Self-Organizing Systems (SASO)*. Ann Arbor, MI, USA: IEEE, pp. 227–234.
16. Gelenbe, E. (2012). Energy packet networks: smart electricity storage to meet surges in demand. In *Proceedings of the Fifth International ICST Conference on Simulation Tools and Techniques*. Sirmione-Desenzano, Italy: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), pp. 1–7.
17. Gelenbe, E. & Bi, H. (2014). Emergency navigation without an infrastructure. *Sensors* 14(8): 15142–15162.
18. Gelenbe, E. & Cao, Y. (1998). Autonomous search for mines. *European Journal of Operational Research* 108(2): 319–333.
19. Gelenbe, E. & Fourneau, J.-M. (2002). G-networks with resets. *Performance Evaluation* 49(1): 179–191.
20. Gelenbe, E., Lent, R., & Nunez, A. (2004). Self-aware networks and QoS. *Proceedings of the IEEE* 92(9): 1478–1489.
21. Gelenbe, E., Lent, R., & Xu, Z. (2001). Towards networks with cognitive packets. In *Performance and QoS of next generation networking*. London: Springer, pp. 3–17.
22. Gelenbe, E., Lent, R., & Xu, Z. (2001). Design and performance of cognitive packet networks. *Performance Evaluation* 46(2): 155–176.
23. Gelenbe, E. & Morfopoulou, C. (2011). A framework for energy-aware routing in packet networks. *The Computer Journal* 54(6): 850–859.
24. Gelenbe, E. & Pujolle, G. (1998). *Introduction to networks of queues*. New York and Chichester: John Wiley Ltd.
25. Gelenbe, E., Xu, Z., & Seref, E. (1999). Cognitive packet networks. In *Proceedings of 11th IEEE International Conference on Tools with Artificial Intelligence, 1999*. Chicago, IL, USA: IEEE, pp. 47–54.
26. Gorbil, G. & Gelenbe, E. (2013). Disruption tolerant communications for large scale emergency evacuation. In *2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. San Diego, CA, USA: IEEE, pp. 540–546.
27. Greer, M. & Ngo, J.W. (2012). Personal emergency preparedness plan (pepp) facebook app: Using cloud computing, mobile technology, and social networking services to decompress traditional channels of communication during emergencies and disasters. In *2012 IEEE Ninth International Conference on Services Computing (SCC)*. Honolulu, HI, USA: IEEE, pp. 494–498.
28. Jimenez-Palacios, J.L. (1998). Understanding and quantifying motor vehicle emissions with vehicle specific power and tildas remote sensing. Ph.D., Massachusetts Institute of Technology.
29. Ke, B.-R., Wang, C.-P., Lin, C.-L., & Wu, Y.-K. (2010). Fuel-efficient driving strategies for highway vehicles. In *2010 the Fifth IEEE Conference on Industrial Electronics and Applications (ICIEA)*. June 2010, Taichung, Taiwan, pp. 2070–2075.
30. Lent, R., Abdelrahman, O.H., Gorbil, G., & Gelenbe, E. (2010). Fast message dissemination for emergency communications. In *Proceedings of PerCom Workshop on Pervasive Networks for Emergency Management (PerNEM'10)*. Germany: Mannheim, NY, USA: IEEE, pp. 370–375.
31. Li, J., Li, Q., Khan, S.U., & Ghani, N. (2011). Community-based cloud for emergency management. In *2011 6th International Conference on System of Systems Engineering (SoSE)*. Albuquerque, NM, USA: IEEE, pp. 55–60.
32. Liebscher, A., Krumnow, M., Krimmling, J., Hanisch, F., & Baker, B. (2015). Energy-efficient routing strategies based on real-time data of a local traffic management center. in *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, June 2015, Budapest, Hungary, pp. 74–80.
33. Onvural, R.O. (1995). *Asynchronous transfer mode networks: performance issues*. Norwood, MA, USA: Artech House, Inc.
34. Osorio, C. & Nanduri, K. (2015). Energy-efficient urban traffic management: a microscopic simulation-based approach. *Transportation Science* 49(3): 637–651.
35. Stepanov, A. & Smith, J.M. (2009). Multi-objective evacuation routing in transportation networks. *European Journal of Operational Research* 198(2): 435–446.
36. Stützle, T. & Hoos, H.H. (2000). Max–min ant system. *Future generation computer systems* 16(8): 889–914.
37. Vreeswijk, J., Mahmod, M., & van Arem, B. (2010). Energy efficient traffic management and control—the ecomove approach and expected benefits. In *2010 13th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, September 2010, Funchal, Madeira Island, Portugal, pp. 955–961.
38. Yao, E., Lang, Z., Yang, Y., & Zhang, Y. (2015). Vehicle routing problem solution considering minimising fuel consumption. *IET Intelligent Transport Systems* 9(5): 523–529.
39. Yao, E. & Song, Y. (2013). Study on eco-route planning algorithm and environmental impact assessment. *Journal of Intelligent Transportation Systems* 17(1): 42–53.

https://doi.org/10.1017/S0269964816000115 Published online by Cambridge University Press