Better safe than sorry: a vehicular traffic re-routing based on traffic conditions and public safety issues

Allan M. de Souza[1,2*] ⬤, Torsten Braun[2], Leonardo C. Botega[3], Raquel Cabral[1], Islene C. Garcia[1]
and Leandro A. Villas[1]

Abstract

Vehicular traffic re-routing is the key to provide better traffic mobility. However, taking into account just traffic-related information to recommend better routes for each vehicle is far from achieving the desired requirements of proper transportation management. In this way, context-aware and multi-objective re-routing approaches will play an important role in traffic management. Yet, most procedures are deterministic and cannot support the strict requirements of traffic management applications, since many vehicles potentially will take the same route, consequently degrading overall traffic efficiency. So, we propose an efficient algorithm named as Better Safe Than Sorry (BSTS), based on Pareto-efficiency. Simulation results have shown that our proposal provides a better trade-off between mobility and safety than state-of-the-art approaches and also avoids the problem of potentially creating different congestion spots.

Keywords: Vehicular traffic re-routing, Intelligent transportation system, Vehicular ad hoc networks

1 Introduction

Vehicular traffic congestion is an emergent problem in many cities around the world. Recent studies [11, 27] have shown that vehicular traffic re-routing is one of the key solutions to handle this problem, which is about suggesting alternative routes to improve the overall traffic efficiency. Google Maps, TomTom and Waze are existing smartphone based Vehicular Navigation Systems (VNS) that handle with traffic congestion by recommending the fastest routes for users by using current traffic information and historical traffic statistics [17].

Considering only traffic-related information for computing better routes to vehicles is far from achieving the requirements of traffic management solutions [32]. In this sense, context-awareness will play an essential role in vehicular traffic re-routing, paving the way for improving not only vehicular traffic mobility, but also other metrics that could not be achieved by just minimizing traffic-related ones, such as driving experience, energy consumption of electric vehicles as well as safety of drivers and passengers [12].

Specifically, the public safety is one of the most concerning issues that needs to be taken into consideration while computing alternative routes, since in some cases it can put the life of drivers and passengers at risk. For instance, a couple that has followed a route recommended by Waze got shot as they crossed a slum of Rio de Janeiro Brazil [7]. Another case, occurred in Boston, USA, when a vehicle, guided by a VNS, crossed a gunfight [25].

Intelligent transportation systems (ITSs) have the potential to enable more efficient, safer and greener transportation, by integrating advanced sensing, processing and communication technologies [21]. With ITSs a whole new set of services will take place, ranging from road safety improvement to traffic efficiency optimization. This vehicular experience will ultimately have a profound impact on society and the daily lives of millions of people around the world, changing the way that we live, work and play. In this context, by integrating different pieces of information available in urban environments,

*Correspondence: allanms@lrc.ic.unicamp.br; souza@inf.unibe.ch
[1]Institute of Computing, University of Campinas, Campinas, São Paulo, Brazil
[2]Computer Science, University of Bern, Bern, Switzerland
Full list of author information is available at the end of the article

ITSs can derive solutions to deal with traffic congestion while improving the safety of drivers and passengers.

Vehicular ad hoc networks (VANETs) [35] and the fifth-generation (5G) networks [13] will play an essential role in supporting ITS applications with vehicular communications ranging from vehicle-to-vehicle (V2V) to vehicle-to-everything (V2X) communications [6, 33]. The infrastructure of VANETs allows the vehicles communicate with each other and with roadside infrastructures, smart objects, in-road, and roadside sensors, and even with people to produce useful information to maximize the potential of ITSs.

The need for context-aware vehicular re-routing is evident. However including a new metric to the re-routing strategy naturally increases its complexity. Bi-objective shortest paths are an instance of the multi-objective shortest paths problem, which is known to be NP-hard, since the number of solutions may be exponential in the nodes number [22]. In this way, several solutions have been proposed to deal with such problems [22] such as Single-Source Shortest Path [22], Weighted-Sum [23] and Resource Constrained Shortest Path (RCSP) [18]. However, most solutions for dealing with such problem are deterministic which are not right solutions for traffic management applications, since many vehicles with the same origin and destination can take the same route, potentially degrading the traffic efficiency.

Unfortunately, most of the literature solutions still only consider traffic-related information to deliver their services [2, 11, 15, 26, 36], which naturally can lead to safety-related issues. On the other hand, navigation systems focusing only on optimal safety [16, 30] can lead to stressful routes, since the recommendation algorithm can include congested roads into the recommended route, but these roads are more likely to be avoided in drivers' criteria during their route planning [1]. Moreover, previous work [8, 9] has shown that using standard multi-objective optimization solutions are not enough to deal with the traffic management problem, since they lead to deterministic solutions that potentially degrade the overall traffic efficiency.

In this way, we propose Better Safe Than Sorry (BSTS) an ITS for vehicular traffic re-routing that considers both traffic conditions and public safety issues while employing a non-deterministic multi-objective routing algorithm to improve the traffic management. BSTS is an extension of our previous work [8], overcoming the limitation of creating different congestion spots during the multi-objective re-routing. In this way, the contributions of this paper are stated as follows:

Non-deterministic multi-objective re-routing algorithm: BSTS implements a non-deterministic vehicular traffic re-routing algorithm based on the Pareto-efficiency (e.g., set of routes to improve both traffic efficiency and safety) to distribute the traffic over the set of feasible routes in polynomial time. This approach reduces the problem of creating different congestion spots during the route recommendation, which is a limitation presented in literature approaches [8, 9, 14, 16, 26].

Analysis of system efficiency and re-routing complexity: The overhead produced by a traffic re-routing system can degrade its overall performance; also the complexity of re-routing algorithm can introduce an undesired latency under denser scenarios, which can degrade even more the system efficiency. Thus, we evaluate the overhead produced by BSTS and the complexity of its re-routing algorithm in respect to literature solutions [9, 14, 16] to evaluate its scalability and whether BSTS can enable a real-time re-routing.

Evaluation of the driver's preference over the safest and the fastest route: Different drivers can have different preferences related to the fastest and the safest routes. For instance, careful drivers may prefer a safer route than a fast one to travel, while drivers with a time limitation potentially prefer a faster route to reach their destination in time. Therefore, we analyze the BSTS performance when drivers choose the fastest and the safest routes. However, to keep the overall improvement in both metrics (e.g., traffic efficiency and mobility), during this analysis BSTS filters the Pareto set to have either the k fastest routes or the k safest routes. Thus, it still can improve not only the traffic efficiency but also safety using the non-deterministic multi-objective re-routing algorithm.

The remainder of this paper is organized as follows. Section 2 describes related work highlighting their advantages and limitations. Section 3 introduces BSTS. Section 4 presents the results. Finally, Section 5 concludes the paper and discusses future work.

2 Related work

In this section, we outline the main related work about vehicular traffic re-routing considering traffic conditions and safety issues.

When we perform a vehicular traffic re-routing, we consider different types of information about the urban environment, such as road geometry, traffic flow, weather conditions, human behavior, traffic light, crash risk, and public safety information [32]. Thanks to VANETs, vehicles can act as mobile sensing nodes that can sense their driving environment to provide real-time traffic-related data using either vehicle-to-vehicle (V2V) or vehicle-to-infrastructure (V2I) communications [19]. On the other hand, to acquire additional pieces of information about the urban environment, different approaches can be considered, including crowd-sourcing based systems, social media, participatory sensing, Web information, etc [9].

The vehicular traffic re-routing problem is still under investigation, most literature solutions considering just

information about traffic conditions to employing re-routing algorithms which is far from achieving the requirements of future traffic management systems [32]. Zhang et al. [37] proposed an application named DIFTOS, that is a distributed infrastructure-less congestion prediction and traffic optimization system for VANETs in an urban environment. DIFTOS presents a hierarchical fashion design that aims at finding for each vehicle the shortest path from its source to the destination point. The first shortest path is computed based on the link travel delay with the weight constraint, after that the algorithm determines a set of predicted congested road segments. Following that, a re-routing procedure is applied to each one of these segments or sub-paths formed by the predicted congested road segments.

Pan et al.[27] proposed a distributed vehicular re-routing system (DIVERT) for congestion avoidance. DIVERT offloads a large part of the rerouting computation at the vehicles, and thus, the re-routing process becomes practical in real-time. The vehicles exchange messages over VANETs, making the re-routing decisions collaborative, use a server and Internet communication to determine an accurate global view of the traffic.

Wang et al. [34] develop a vehicle re-routing system called Next Road Rerouting (NRR), where a heuristic re-routing decision considers a cost function that takes into account the driver's destination and local traffic conditions. Also, apply periodic re-routing algorithms.

As a result of the rapid urbanization of cities and consequently the increasing number of vehicles, some problems arise in addition to congestion, such as pollution, traffic accidents, and unsafe areas. In this sense, several papers have been combining different types of information to optimize traffic planning. Peng et al. [29] discuss the main issues to conduct real-time road safety prediction and propose a new deep learning framework (DeepRSI) to solve this problem. In this work, the authors use public data from authoritative official organizations in New York City, that include urban maps, weather data, holiday event data, GPS trajectories generated, and accident event records. Souza et al. [11] introduce ICARUS, a distributed and pro-active Traffic Management System, which receives notifications about traffic events (congestion detection, accident notification, and congestion prediction) then it can calculate new routes, and, later, notify drivers to follow new paths pro-actively by using inter-vehicle communications. Doolan and Muntean [15] proposed an eco-friendly ITS (EcoTrec) which intends to reduce carbon emissions and to improve traffic efficiency. In EcoTrec, there is a traffic model built by a central entity which contains the average fuel consumption of each road. All vehicles periodically report their position and average fuel consumption to the central entity using an "endemic routing" data dissemination protocol to ensure the

delivery. In this way, all vehicles consult this traffic model periodically to reroute themselves. EcoTrec uses the Dijkstra algorithm to compute optimal routes to each vehicle, which does not assure traffic efficiency.

Another way to improve traffic conditions is the Vehicular Navigation System (VNS). In most cases, this system calculates the route once and using the shortest travel delay path algorithm based on traffic statistics or real-time road traffic measurement. In this case, the routes that they provide are individually optimized paths for each vehicle and the congestion is detected based on vehicles speed in different road segments [37]. For example, Google Maps, TomTom and Waze are navigation systems to suggest faster routes to users, based on overall traffic knowledge [17]. Sometimes, trying to avoid traffic congestion, these applications can guide passengers to unsafe areas, where there are cases of shootings, assaults, and even murder [7, 25].

Criminal data has already been used to determine safe routes in [16, 30]. Galbrun et al. [16] were the first to define safe routes with criminal data. They used public datasets of criminal activity, as well as on city-dwellers mobility traces to develop a risk model for an urban road network from Chicago and Philadelphia. So, they introduced SafePaths and modeled it as bi-objective shortest path problem. They have used a deterministic algorithm to determine the better route, by computing all possible routes. Their work presents two limitations: *(i)* computing all possible routes can be very costly, potentially degrading the system performance; *(ii)* in the case of a driver who is just considering the distance aspect, such an approach will not provide better traffic management, because traffic conditions on the roads are not known. Shah et al. [30] proposed a crowd-sourcing-based system to suggest the safest routes for the drivers. To do this, they use data (reported by the people using their smartphones) about crime-related incidents. However, as its mechanism is based only on crowd reports, so the information is not always correct and sufficient to provide traffic efficiency. In earlier work [9] we proposed an Intelligent Transportation System for improving SAfety and traFfic Efficiency named as itsSAFE, which: *(i)* employs accurate knowledge about the traffic conditions and unsafe levels on roads; *(ii)* and use these data to suggests periodically alternative routes for all vehicles. The problem is modeled as an instance of the Resource Constrained Shortest Path (RCSP) and minimizing two objectives (traffic congestion and unsafety level). A dynamic programming routing algorithm is used to determine the most efficient route to a vehicle satisfying the safety requirements.

Table 1 shows a qualitative comparison between BSTS and the literature solutions. It presents a summary of literature solutions highlighting the characteristics that we

Table 1 Summary of literature solutions highlighting their characteristics and limitations

Referred papers	Architecture			Re-routing strategy		Optimization			Re-routing considerations		
	Centralized	Distributed	Hybrid	Deterministic	Non-deterministic	Single-objective	Multi-objective	Traffic flow	Traffic accident	Pollution	Safety issues
BSTS			✓		✓		✓	✓			✓
itsSAFE [9]	✓			✓			✓	✓			✓
VNS Systems	✓			✓		✓		✓	✓		
DIFTOS [37]		✓		✓		✓		✓			
DeepRSI [29]	✓			✓			✓	✓	✓		
DIVERT [27]			✓		✓	✓		✓			
EcoTrec [15]			✓	✓		✓				✓	
NRR [34]	✓			✓		✓		✓	✓		
ICARUS [11]	✓				✓	✓		✓	✓		
SafePaths [16]	✓			✓			✓				✓
Crowdsafe [30]	✓			✓		✓					✓
SAFER [10]		✓		✓		✓		✓			

Bold solutions are used to validate our approach

consider in this work: architecture re-routing, strategy optimization, re-routing considerations. Additionally, we stand out the solutions used to validate our approach, itsSAFE, VNS Systems, EcoTrec, and SafePaths. We can see in Table 1 that most of the solutions consider just traffic flow to do re-routing and few works consider extra information, like a traffic accident, pollution, and safety issues. In particular, only three papers examine the safety issues data, so it is a relevant aspect to consider to planning routes.

3 Better safe than sorry

BSTS is designed based on the following major requirements. First, the route planning must be offloaded from the cloud to the vehicles, to minimize computation effort and communication cost, consequently improving the system scalability. Second, the system needs to have access to public safety information of the city to extract knowledge about risk areas. At last, the route planning algorithm needs to be able to balance the traffic flow over the roads to avoid creating bottlenecks in the transportation infrastructure (e.g., traffic congestion) while dealing with safety issues.

3.1 Urban scenario modeling

Let $D = (V, E)$ be a digraph that represents the road network, in which the set of vertices V represents the intersections, while the set of edges $E \subseteq V \times V$ corresponds to the roads, e.g., an edge $uv \in E$ corresponds

to a road connecting intersection u to intersection v, for $u, v \in V$. Each road $uv \in E$ has two attributes, $\tau_{uv} \geq 0$ and $r_{uv} \geq 0$ that represent the traffic condition and the safety risk in road uv, respectively. Also, let N be the set of vehicles in the road network. Each vehicle $n \in N$ has a pair of origin $s \in V$ and destination $t \in V$, such that $s \neq t$, and is associated to a path $P \subseteq E$ connecting s to t. The traffic efficiency of a path is defined by $\tau_P = \sum_{uv \in P} \tau_{uv}$ and its risk level by $r_P = \sum_{uv \in P} r_{uv}$.

3.2 System design

Figure 1 depicts the system architecture, which is composed of vehicles, safety information providers, edge servers (e.g., fog nodes) and a remote cloud. Each vehicle equipped with on-board units (OBU) can communicate with other vehicles, roadside infrastructures (e.g., RSUs, 5G base stations), edge servers, and with the cloud using either 5G or VANET communications to report traffic-related information and to receive updates about traffic conditions and safety risks for improving their mobility and security. On the other hand, edge servers widely deployed on roadside infrastructures can provide processing and storage resources on the network edge, consequently, enabling fast responsiveness [21]. In this sense, each fog node is responsible for building local pieces of knowledge about traffic conditions over its coverage based on vehicle's report. Naturally, the remote cloud is responsible for creating global awareness about traffic conditions and risk areas and
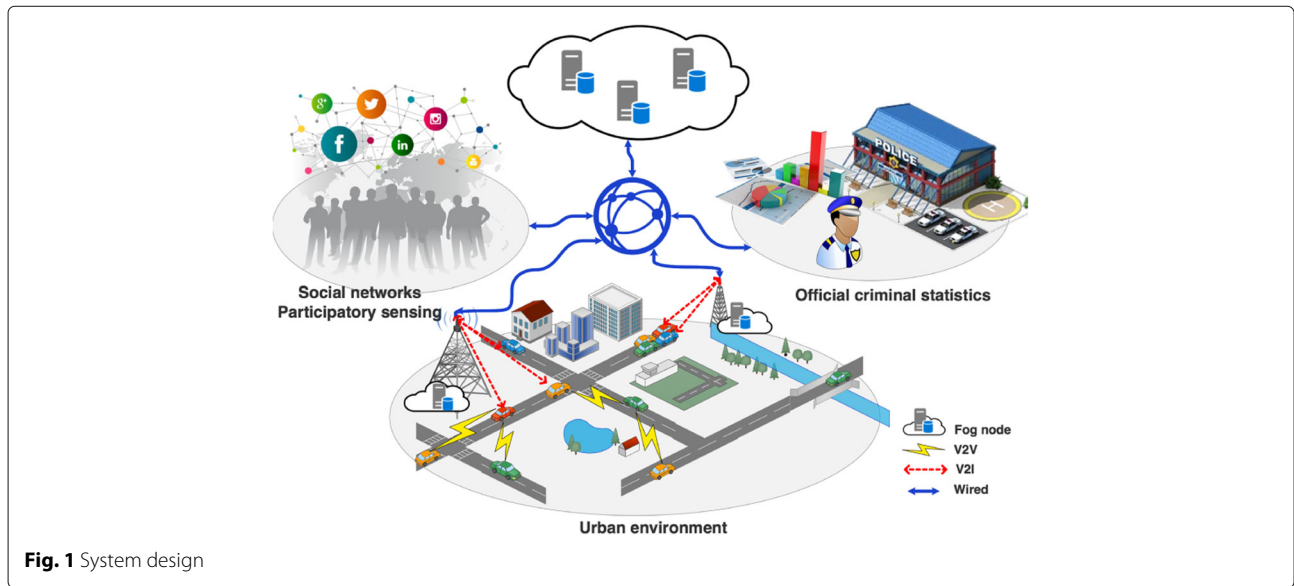
Fig. 1 System design

also to deal with resource constraints presented by edge servers. Finally, safety information providers are responsible for feeding the system with criminal reports to build pieces of knowledge about city-wide illegal activities. Those providers include *(i)* police departments providing the historical data of official crime statistics; and *(ii)* people providing a more dynamic knowledge and real-time sensing based on crowd-sourcing approach (e.g., participatory sensing applications and social media), which is not possible by using just historical data.

In this way, the system can build pieces of knowledge about traffic conditions and dangerous areas over the city. Thus, upon detecting signs of congestion, the system can send a notification to the vehicles, consequently enabling them to plan a route for improving their mobility and the safety of driver and passengers.

3.3   Traffic information reporting
The system needs to receive information about every vehicle in the scenario to create accurate knowledge about traffic conditions. However, if each vehicle reports its information periodically, a considerable number of transmissions will be produced, potentially overloading the communication channel, and consequently, degrading overall system performance. In this way, BSTS employs an efficient mechanism to reduce the number of transmissions and still maintaining accurate knowledge of traffic conditions. To do so, BSTS uses the approach described in [10], in which the authors segment the road network into sub-regions based on the size of vehicles' communication range, and define a centroid to each sub-region. Figure 2 illustrates the road network segmentation and the traffic information reporting employed by BSTS. In summary, we can see two different sub-regions with their respective centroids (e.g., black circle in the center of the region) and vehicles. After building the traffic knowledge of its sub-region, each vehicle schedules a report based on the distance to the centroid (represented by the arrows linking the centroid to the vehicles in the sub-region). Using this approach, the vehicle closest to the centroid will have shorter delay than further vehicles. Thus, it will report the traffic information (represented by the red vehicle and information reported by it) first than the other vehicles, meanwhile, if another vehicle hears that report it will cancel its report to avoid redundant transmissions.

Each vehicle within a sub-region shares its traffic-related information with nearby vehicles using beacon messages. This information is: *(i)* current position given by its road $uv \in E$; and *(ii)* speed $speed_n$. Therefore, all vehicles in the same sub-region know the number of vehicles in each road (within its communication range) and speed of each one; thus each vehicle can build a piece of knowledge of traffic condition of its sub-region [10].

To reduce the number of transmissions, BSTS implements a delay-based approach, which each vehicle within the same sub-region schedules the transmission of its piece of traffic knowledge based on the distance between its current position and the centroid of its sub-region. This mechanism prioritizes vehicles closer to each centroid, and when the first transmission of a sub-region occurs, all vehicles within the same sub-region that listened to this transmission will cancel their scheduled transmission. Figure 2 describes this procedure.

3.4   Traffic condition estimation
With the traffic knowledge of each sub-region, BSTS computes the average speed of each road uv periodically (during the re-routing phase based on the re-routing
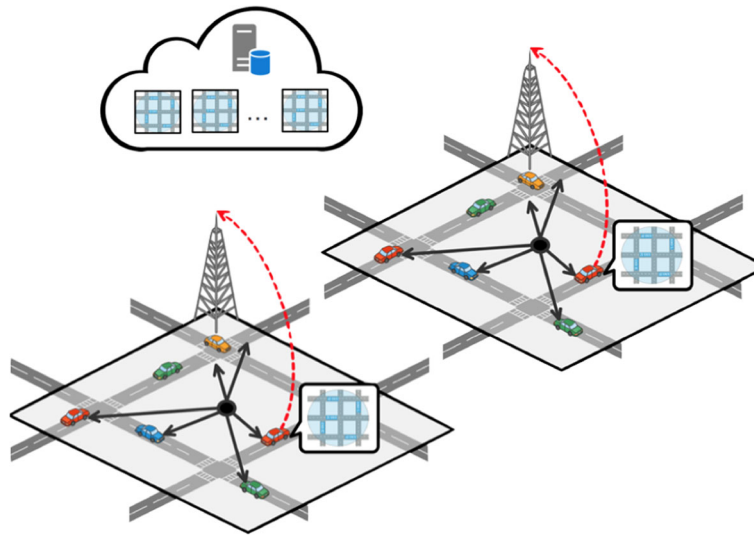
Fig. 2 Sub-regions employed by BSTS

frequency) using the information about the number of vehicles on it and their respective speeds $speed_n$ received in the last report. Thus, the average speed of each road $avg_s(uv)$ at time t is:

$$avg_s^t(uv) = \frac{\sum_{n \in N_{uv}^t} speed_n^t}{|N_{uv}^t|}, \quad (1)$$

where N_{uv}^t is the set of vehicles on the road uv at time t. Then, based on the average speed of each road, BSTS computes its traffic condition τ_{uv} at time t, defined as:

$$\tau_{uv} = \frac{max_s(uv) - avg_s^t(uv)}{max_s(uv)}, \quad (2)$$

where $max_s(uv)$ is the maximum speed of the road uv. When the system does not receive any information from a specific road, that road is defined as free-flow; thus, its mean speed is the same value of its maximum speed.

Based on τ_{uv} it is possible to detect bottlenecks in the traffic efficiency, using the Level of Service (LOS) classification defined by the Highway Capacity Manual (HCM) [4]. The relation of τ_{uv} and its classification is described in Table 2.

Table 2 Relation of τ_{uv} and its classification

| τ_{uv}	LOS	Traffic classification
(0, 0.15]	A	Free flow
(0.15, 0.33]	B	Free flow
(0.33, 0.50]	C	Slight congested
(0.50, 0.60]	D	Slight congested
(0.60, 0.70]	E	Congested
(0.70, 1.00]	F	Congested

3.5 Safety risk estimation

To estimate the safety risk of each road, BSTS uses information about criminal activities over the city. This information is provided by heterogeneous sources, being: State Departments of Public Safety, social media (e.g., Twitter), participatory sensing applications (e.g., Onde Fui Roubado - Where I Was Robbed) and news websites that report crime events. The first one provides official data (e.g., crime records) about city crimes, generally considered as the most reliable source, and the other ones offer non-official data, usually unreliable. However, non-official data are considered a valuable qualitative source of human-provided input.

It is expected that official data to be better organized and having more significant information. However, it also might have missing and information (e.g., blank fields), which can make the correct identification and quantification of crime events harder. Moreover, most criminal records are unstructured, since it is often provided by victims who still are emotionally affected by the criminal event, consequently directly influencing in the data quality.

To properly analyze crime data (official and non-official), improve their quality and infer specific crime events, a previously developed assessment process was applied. Such process is composed by pre-processing to clean and discard dirty data, data mining methods for extracting relevant objects, fusion techniques for correlation and integration, and semantic models to obtain meaningful information. The result of this assessment process is a reliable criminal event database[5].

Therefore, to each processed data sample (official and non-official) BSTS employs mechanisms to *(i)* extract the

Fig. 3 Safety risk estimation data flow
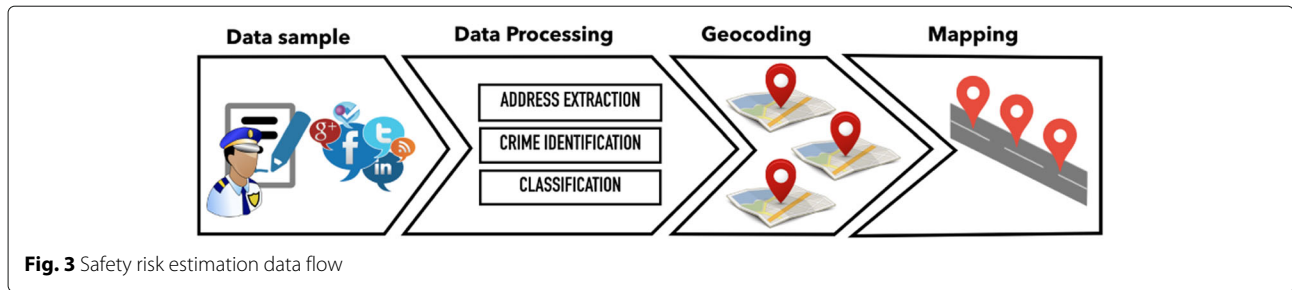
information about the location of each criminal event using its address (e.g., geocoding); *(ii)* map the location of each crime to the closest road; and *(iii)* characterize the safety risk of each crime based on its complexity (e.g., robbery, theft, extortion abduction, etc.).

In this way, it is possible to classify each criminal event attributing a weight based on its characteristics [5]. Figure 3 depicts the data flow for estimating a safety risk of each road. Besides, Table 3 describes how each criminal activity is classified, presenting its type, a brief description, and its safety risk.

To map criminal events to the closest road, BSTS uses the coordinates obtained from the address of the criminal activity as well as the coordinates of all vertices from the road network. In this way, it is possible to identify which vertices $u, v \in V$ are closest to the criminal event, such that $uv \in E$.

Upon classifying and mapping each criminal activity, BSTS computes the safety risk of each road r_{uv} based on the weighted-sum of crimes happened on each road, as follows:

$$r_{uv} = \sum_{c \in CE_{uv}} risk_c, \qquad (3)$$

where CE_{uv} is the set of criminal events within the road uv and $risk_c$ is the safety risk of each criminal event c.

3.6 Improving mobility and safety

When BSTS detects signs of congestion (e.g., $\tau_{uv} \geq 0.6$ at time t), it sends a re-routing notification containing the traffic conditions and the safety risk estimations to the closest vehicle to the congestion that has sent the last traffic estimation reporting. Therefore, this vehicle is responsible for disseminating the message to its neighbors. To ensure that all vehicles that are going towards the congestion receive the message, BSTS employs a multi-hop data dissemination protocol that addresses the broadcast storm problem using a delay-based broadcast suppression mechanism, whose the forwarding delay is based on the inverse distance between sender and receiver [11]. In this way, as a vehicle receives the re-routing notification, it can compute a new route for improving its mobility and safety.

BSTS implements a non-deterministic optimization algorithm to improve both metrics (e.g., mobility and safety) and also to minimize the problem of creating different congestion spots which are inherent to deterministic approaches [14, 26]. To do so, BSTS defines a set of potential routes and then selects one of them using a probabilistic approach. The set of potential routes is determined based on a Pareto curve, which provides the fastest route (e.g., optimal) to every possible safety risk value of each route. Figure 4 shows an example of a Pareto curve to optimize both mobility and safety. In this example, the safer the route is, the slower it becomes.

The complete re-routing algorithm is described in Algorithm 1. First, as soon as a vehicle receives the re-routing notification, it updates the both pieces of knowledge (Line 1) and computes the waiting time to forward the message to nearby vehicles (Line 2). The knowledge update is performed locally in each vehicle using a linear function based on the size of the road network graph D with complexity $O(E)$, this update is done whenever a vehicle receives a knowledge update with a re-routing notification (e.g., in our implementation this notification has a frequency of 450 s). Thereafter, the vehicle computes the Pareto set $\mathcal{P}$ based on its current position s and destination t (Lines 3-6). It is important to consider that different drivers can have different preferences concerning a safer or faster route. Thus, we enabled that each vehicle filters the Pareto set according to its preference. However, to maintain a suitable traffic flow balancing the filtering is based on the set of k fastest or k safest routes (Lines 8 and 11). Then, each vehicle applies the Boltzmann

Table 3 Criminal activity classification

Type	Criminal event description	Risk
1	Low complexity events without fire guns executed by a single individual	1
2	Medium complexity events potentially executed by two individuals having fire guns in more complexity locations through a planned schema	2
3	High complexity events executed by several individuals having fire guns and/or especial weapons in high complexity locations	4

Algorithm 1: Whenever a vehicle receives a re-routing notification from the server

---

   Input : D // *road network with updated values of* τ_{uv} *and* r_{uv} $\forall uv \in E$
       preference // *preference of selecting the set of fastest or safest routes from the Pareto set*

   Output: Alternative route for improving mobility and safety

1   D.updateKnoweledge();
   // Compute the delay to forward the message
2   $T \leftarrow computeDelay()$;
3   $s \leftarrow$ currentPosition;
4   $t \leftarrow$ destination;
5   **if** $s \neq t$ **then**
      // Compute the Pareto set from s to t
6     $\mathcal{P} \leftarrow$ Pareto(s,t);
      // Check the re-routing preference
7   **if** *preference* $==$ *Fastest* **then**
8       $\mathcal{P} \leftarrow$ getKFastestRoutes$(\mathcal{P}, k)$;
9   **end**
10   **else**
11      $\mathcal{P} \leftarrow$ getKSafestRoutes$(\mathcal{P}, k)$;
12   **end**
      // Select a route from the Pareto set using the Boltzamann distribution
13   *new_route* $\leftarrow$ BoltzmannSelection$(\mathcal{P})$;
14   setNewRoute(new_route);
15   **end**
   // wait for T seconds
16   wait(T);
17   **if** *did not listen to a re-transmission in T seconds?* **then**
18   forwardMessage();
19   **end**
20   **else**
21   Cancels the transmission;
22   **end**

Algorithm 2: Dynamic programming algorithm for computing the Pareto set.

   Input : s // *current position of vehicle n*
        t // *destination of vehicle n*
        λ // *maximum risk of the path*

   Output: Pareto set of paths stating in s and ending in t with risk at most λ.

1   $DP \leftarrow [\ ]$; // table to store $T(v, \varphi)$
2   $\psi \leftarrow [\ ]$; // stores predecessor vertex in the route $DP[v, \varphi]$
3   **foreach** $\varphi \in \{1, 2, \ldots \lambda\}$ **do**
4    $\psi[s, \varphi] \leftarrow \emptyset$;
5    $DP[s, \varphi] \leftarrow 0$;
6   **end**
7   **foreach** $v \in V \setminus \{s\}$ **do**
8    $\psi[v, 0] \leftarrow \emptyset$;
9    $DP[v, 0] \leftarrow \infty$;
10   **end**
11   **foreach** $\varphi \in \{1, 2, \ldots, \lambda\}$ **do**
12   **foreach** $v \in V$ **do**
13     $\psi[v, \varphi] \leftarrow \psi[v, \varphi - 1]$;
14     $DP[v, \varphi] \leftarrow DP[v, \varphi - 1]$;
      // Edge linking v to its predecessor u
15   **foreach** $uv \in E$ **do**
16   **if** $r_{uv} \leq \varphi$ **then**
17   **if** $[v, \varphi] > DP[u, r_{uv} - 1] + \tau_{uv}$ **then**
18        $\psi[v, \varphi] \leftarrow u$;
19        $DP[v, \varphi] \leftarrow DP[u, r_{uv} - 1] + \tau_{uv}$;
20   **end**
21   **end**
22   **end**
23   **end**
24   **return** $DP[t,], \psi$;
25   **end**

---

algorithm [20] to select a route from the filtered Pareto set (Line 13), which will be further defined as its new route (Line 14). If the vehicle did not hear any transmission of the update message in T seconds, it forwards its message; otherwise, it cancels the transmission.

3.7 Computing the pareto set

The problem to find the Pareto set of with origin s and destination t is similar to the problem of finding the fastest path P with $r_P \leq \varphi$ such that $\varphi \in \{1, 2, \ldots, \lambda\}$, thus, satisfying the following recurrence:

$$T(v, \varphi) = \begin{cases} 0 & \text{if } \varphi = 0 \text{ and } v = s, \\ \infty & \text{if } \varphi = 0 \text{ and } v \neq s, \\ \min \begin{cases} T(v, \varphi - 1) \\ \min\limits_{u: r_{uv} \leq \varphi} \{T(u, r_{uv} - 1) + \tau_{uv}\} \end{cases} & \text{Otherwise} \end{cases} \quad (4)$$

Such recurrence naturally derives a recursive algorithm with exponential time complexity, but it also can be solved using a dynamic programming approach having complexity $O(E\lambda)$, which has polynomial complexity since, in our scenario, the value of λ is not arbitrarily large.
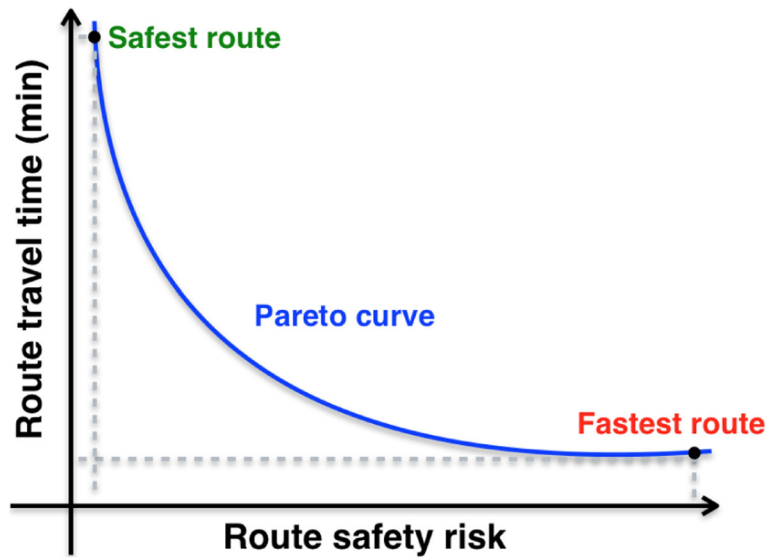
Fig. 4 Example of a Pareto curve for mobility and safety

Algorithm 2 describes the dynamic programming approach used for computing the Pareto set. In the algorithm, *DP* is a dynamic programming table used to memorize the traffic efficiency value to reach each vertex v from a vertex s with risk φ, for $s, v \in V$. Such a path can be stored by table ψ, in which $\psi[v, \varphi]$ is a predecessor of v in the (s, v)-path with risk φ. The first and the second loops (lines 3-10) are responsible for preparing the *DP* and ψ tables in the base cases of the recurrence, which mark empty paths by clearing the predecessor of source s, and mark (s, v)-paths as unfeasible whenever the demanded risk is $\varphi = 0$. Later, the algorithm fills up the table *DP*, for the following risk values $\varphi \in \{1, 2, \ldots, \lambda\}$: the best traffic efficiency is either the same as that of risk $\varphi - 1$ (line 14), or is achieved by following some non empty (s, v)-path from s to some predecessor u, and edge uv (lines 15-22). Possible predecessors are those vertices u for which $r_{uv} \leq \varphi$, and the corresponding (s, v)-path in this case has traffic efficiency $DP[u, r_{uv} - 1] + \tau_{uv}$ (line 19). Finally, the Pareto set can be obtained through line t for all values of φ.

Figure 5 shows a digraph $D = (V, E)$ with edges having two weights in the form τ_{uv}/r_{uv} representing traffic efficiency and safety risk, respectively, in a certain scenario. Table 4 exhibits the values obtained for *DP* when we run the Algorithm 2 in this example scenario, if a vehicle has origin s and destination t, with safety risk of the shortest
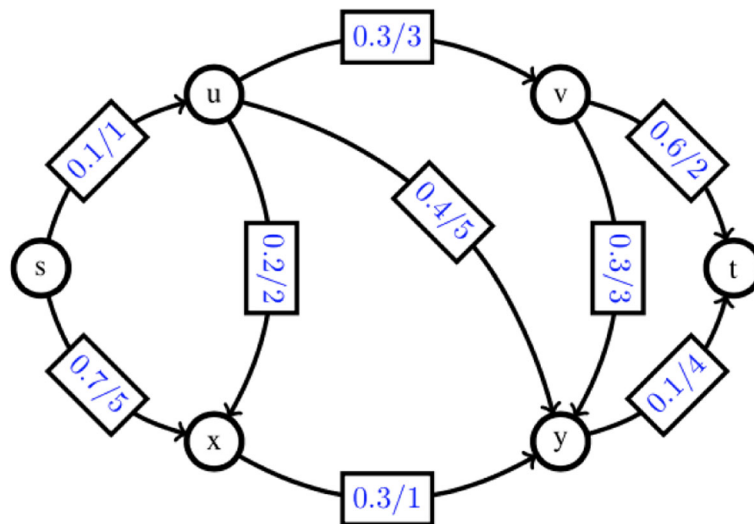

Fig. 5 Digraph $D = (V, E)$ with traffic condition τ_{uv} and safety risk r_{uv} representation built by BSTS

Table 4 Dynamic Programming table for the Digraph of Fig. 5

$v\backslash\varphi$	0	1	2	3	4	5	6	7	8	9	10
s	0	0	0	0	0	0	0	0	0	0	0
u	∞	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
x	∞	∞	∞	0.3	0.3	0.3	0.3	0.3	0.3	0.3	0.3
v	∞	∞	∞	∞	0.4	0.4	0.4	0.4	0.4	0.4	0.4
y	∞	∞	∞	∞	0.6	0.6	0.5	0.5	0.5	0.5	0.5
t	∞	∞	∞	∞	∞	∞	1.0	1.0	0.7	0.7	0.6

path of $\lambda = 10$. It is important to stress that the Pareto set for the scenario mentioned above is $DP[t,]$ (e.g., the whole line).

3.8 Route selection

The Boltzmann selection works as follows. Let k and T be the two parameters used by the Boltzmann algorithm, in which the first one represents the Boltzmann constant and the other one the parameterized attribute. In this way, each vehicle computes its Boltzmann constant according to:

$$k(P) = \sum_{uv \in P} \exp\left(-\frac{\tau_{uv}}{T}\right), \qquad (5)$$

where P is a route from the Pareto set $\mathcal{P}$, τ_{uv} is the traffic condition of each road in the path and T is the parameterized value of the Boltzmann algorithm. Notice that the greater the value of T is, the higher is the chance of achieving a uniform distribution, which must be avoided to provide a better traffic balance (e.g., avoid selecting either a longer alternative path or a least efficient one). It is worth noticing that considering the Pareto set to select the new route for a vehicle we ensure that all routes in that set are improving both metrics (e.g., mobility and safety). Therefore, to provide a better traffic flow balancing, the route selection is based only on the traffic condition of the road τ_P, but the new route will also improve the safety of the drivers and passengers of the vehicle compared its previous route.

After computing the Boltzmann constant, each vehicle can compute the selection probability of each $P \in \mathcal{P}$ based on the traffic condition of the whole path. The probability is defined by $Pr(P)$:

$$Pr(P) = \frac{1}{\exp\left(\frac{\sum_{uv \in P} \tau_{uv}}{k(P) \cdot T}\right)} \qquad (6)$$

where, P is a possible alternative path and $\sum_{uv \in P} \tau_{uv}$ represents its traffic condition.

Finally, based on the probability of each path, the vehicle can choose its alternative route P' using a random procedure based on each probability. The key idea is to select the path with the highest probability that satisfies the following condition:

$$P' = \underset{P \in \mathcal{P}}{\arg\max} \{X \cdot Pr(P), X \in [0, 1]\} \qquad (7)$$

where, X is a random variable to represent the random procedure and can assume values between 0 and 1.

4 Results

This section describes the BSTS performance. First of all, Subsection 4.1 introduces the simulation platform, presenting the tools, the scenario, and the analyzed metrics. Subsection 4.2 describes the dataset of criminal incidents used to extract safety knowledge. Subsection 4.3 presents the network performance analysis. Subsection 4.4 shows the system efficiency and re-routing complexity analysis for BSTS and literature solutions. Subsection 4.5 shows trade-off of traffic efficiency and safety risk of BSTS and literature solutions. Finally, Subsection 4.6 shows the impact of selecting the fastest and the safest routes in BSTS.

4.1 Methodology

To provide the simulation, we use simulator of urban mobility SUMO [3] version 0.30.0, the network simulator OMNeT++ [24] version 5.0 and also the vehicular networking framework Veins [31] 4.6. The road network is composed of a fragment of 5 km^2 from Sao Paulo city, Brazil, obtained using OpenStreetMap. The traffic mobility was produced using the TrafficModeler [28] tool to ensure a close to real traffic mobility resulting in a total of five thousand routes. The number of routes was defined to create free-flow mobility and heavy traffic congestion (i.e., the travel time for the majority of vehicles is significantly higher than the free flow travel time). In this way, we generate the traffic at a constant rate by deploying one car each second in the simulator from one side of the scenario to another one. By default, the shortest travel time paths are automatically calculated and assigned to each vehicle at the beginning of simulation based on the road speed limits. The results are the average over ten runs, which is sufficient because the result variation between the runs is not significant. Table 5 shows additional parameters used in the simulation.

4.2 Safety dataset

The knowledge about safety risks was built based on the official criminal statistics of police departments of the Sao Paulo city, Brazil. The dataset is composed of several criminal reports from 2002 to 2016. However, to this work, just criminal reports related to robbery, theft, extortion abduction and kidnapping of the last three years were used. The time window definition (e.g., 2014 − 2016), was used to minimize issues related to poor data quality,

Table 5 Simulation parameters

Parameters	Values
Channel frequency	5.890e9 Hz
Propagation model	*Two ray*
Transmission power	2.2 mW
Communication range	300 m
Maximum data message size	250 Kbytes
Bit rate	18 Mbit/s
Max hop count	10
Density	100, 250, 500 and 1000 vehicles/km^2
# Sub-regions	18
Re-routing frequency	450 s

for instance, out of date information. Some of the main attributes regarding each crime report include the address of the crime occurrence, its type, and a brief description.

Figure 6 depicts an example of both pieces of knowledge employed by BSTS over the simulated scenario. The left-hand side figure represents the piece of knowledge about traffic conditions, while the other one represents the piece of knowledge about safety risks on the roads.

4.3 Network cost analysis

This section evaluates the network cost and the scalability of each solution by analyzing the overhead produced by them to deliver their services. To this evaluation we have compared BSTS with our previous solution itsSAFE [9] that employs a centralized architecture and also with EcoTrec [15] that employs a hybrid architecture. In this way, we analyze the following metrics:

- **Transmitted messages** are the total number of data messages transmitted by all vehicles in the network during the whole simulation. A high number of

transmissions is a strong indication of redundant communications, which might produce network contentions that can limit the system performance.

- **Accuracy of the traffic view** is the percentage of correct traffic condition estimations computed by the system. The accuracy is measured considering the estimation produced by the system at time t and the real traffic condition in the same timestamp which can be retrieved from the mobility simulator. The accuracy is directly related to the performance of the re-routing algorithm, since the higher the accuracy is the better the re-routing traffic management provided by the system.

Figure 7 shows the results for the analyzed metrics with respect to the density of vehicles comparing BSTS with the literature solutions itsSAFE [9] and EcoTrec [15]. As expected, itsSAFE presents the higher number of transmissions for all densities of vehicles due to its centralized approach. Therefore, every vehicle in the road network needs to report its traffic information to the central server at least once during each re-routing phase. Also, each vehicle needs to request a new route to the central server to improve its mobility, which increases even more the number of V2I transmissions (see Fig. 7a). The centralized architecture employed by itsSAFE produces high overhead, which potentially limits the system scalability. Thus, as the density of vehicles rises the number of transmissions goes up dramatically. However, the centralized architecture contributes to a better knowledge about the traffic conditions, since it the server receives the traffic reports from all vehicles, consequently producing a very accurate knowledge about traffic conditions, which is roughly 100% for all densities of vehicles (see Fig. 7b).

In EcoTrec, a hybrid approach is employed to offload the re-routing computation to each vehicle to enable real-
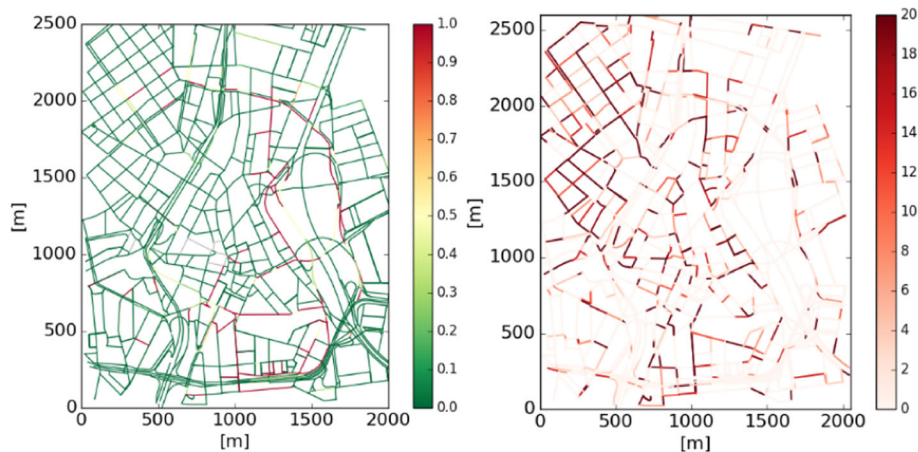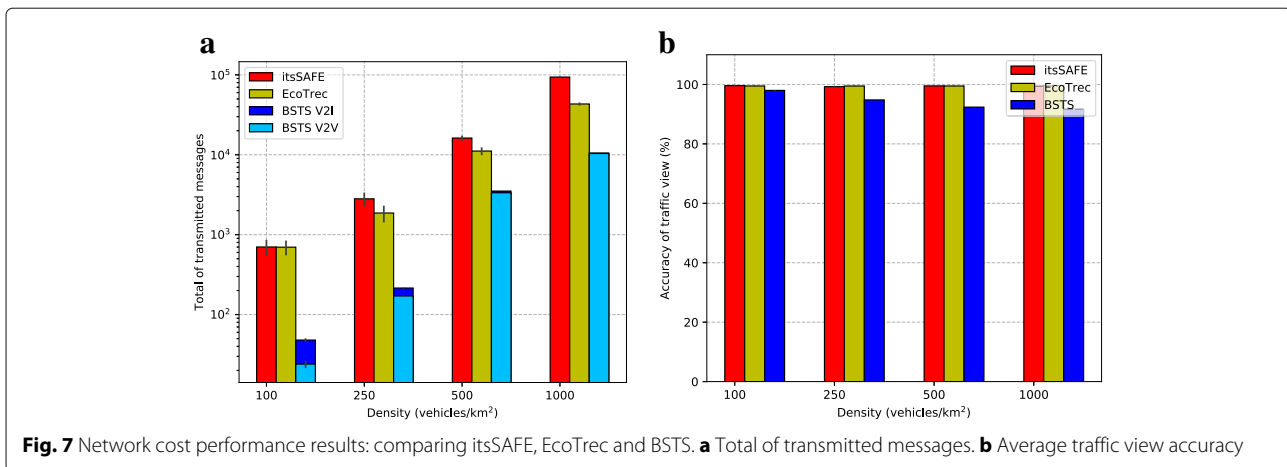

Fig. 6 Pieces of knowledge employed by BSTS

Fig. 7 Network cost performance results: comparing itsSAFE, EcoTrec and BSTS. **a** Total of transmitted messages. **b** Average traffic view accuracy

time traffic management. The offloading approach also reduces the number of transmissions, because the vehicles do not have to spend a new transmission to request a new route to the server. However, EcoTrec does not implement any mechanism to reduce the number of traffic reporting transmissions to build the knowledge about traffic conditions on the roads, thus every vehicle will transmit a traffic report at least once during each route phase. In this way, the reduction in the number of transmissions provided by offloading mechanism implemented by EcoTrec is approximately 50% in higher densities of vehicles, when compared to the number of transmissions achieved by itsSAFE (see Fig. 7a). Nevertheless, EcoTrec produces very accurate traffic knowledge with average accuracy higher than 98%, which is because the server knows the position and velocity of all vehicles in the road network.

The number of transmissions of BSTS is displayed in Fig. 7a as V2I transmissions (with the dark blue color) and V2V transmissions (with light blue color), since BSTS uses both types of transmissions to build the traffic knowledge as well as to spread it out to the vehicles. The effectiveness of the traffic reporting mechanism employed by BSTS can be seen in both results (total number of transmitted messages and traffic view accuracy, respectively). Regarding the number of transmissions, BSTS reduces the total number of transmissions by 90% and 75% when compared to itsSAFE and EcoTrec (see Fig. 7a), respectively. Such a reduction in the total number of transmissions is achieved because only one vehicle per sub-region estimates and reports the traffic information of the entire sub-region to the central server (i.e., V2I transmission).

The majority of the messages transmitted by BSTS is to spread the traffic view to all vehicles (e.g., V2V transmissions) since the knowledge needs to be forwarded throughout a multi-hop approach. The accuracy of the traffic view, although few vehicles report

the traffic information, BSTS still can build an accurate knowledge about traffic conditions with an accuracy greater than 93%, which is the consequence of its efficient mechanism to estimate the traffic in each sub-region. Eventually, the considerable reduction in the number of transmissions presented by BSTS contributes to improve system scalability, despite the increase in the density of vehicles BSTS still can maintain a relatively small number of transmissions when compared to the other solutions.

With the presented results we can conclude that BSTS is a more efficient solution because it presents lower overhead with accurate traffic view. Furthermore, BSTS provides better system scalability in comparison with itsSAFE and EcoTrec since the high network traffic produced by them limits the system scalability. Thus, depending on the density of vehicles, it can overload the network, and consequently, introduces an undesired latency to the system, potentially degrading its overall performance.

4.4 System efficiency and complexity evaluation

To evaluate the overall system performance and its complexity, the following metrics were assessed:

- **System efficiency** is a metric to analyze how efficient each system is. It is computed as the ratio of the traffic density and the number of transmitted messages required by the system to deliver its service. The higher the values is the better the system efficiency will be.
- **CPU time** is the time spent by the re-routing algorithm to compute the new alternative route to the intended vehicles. A longer CPU time potentially introduces latency to the system, consequently degrading its overall performance.

For the complexity analysis, we assume that the shortest path implementation of EcoTrec uses the Fibonacci heap, thus it has a complexity of $O(E + V \log V)$. On the other hand, itsSAFE uses a dynamic programming algorithm with complexity $O(E\lambda)$, in which λ is the upper bound value for the safety risk of a path linking the origin to the destination. Thus, as itsSAFE uses a centralized architecture it will perform such algorithm for the set of vehicles that needs to be re-routed (lets define this set as N_r). In this way, the overall complexity of itsSAFE is $O(N_r E\lambda)$. Eventually, like the itsSAFE, BSTS also implements a dynamic programming based algorithm to compute the Pareto set with complexity $O(E\lambda)$. However, as BSTS offloads the routing computation in each vehicle such algorithm is distributed over the set of vehicles to be re-routed, consequently the overall complexity of BSTS is upper bounded from above by the Pareto set computation complexity, which is just $\mathcal{O}(E\lambda)$.

To evaluate the time complexity, we considered the CPU time of each solution. For this evaluation, we counted the amount of time spent in each re-routing step using an Intel(R) Core(TM) i5-5257U CPU with 2.70 GHz. Figure 8a shows the results for the average CPU usage. As it was expected, itsSAFE presents the highest CPU time for all re-routing steps, which is a consequence of its centralized architecture. Therefore, the server needs to compute the alternative routes for all intended vehicles. In contrast, EcoTrec presents the lowest CPU time not only as a result of its shortest path implementation but also due to its offloading approach. In turn, BSTS also offloads the routing computation in each vehicle, which dramatically reduces the average CPU time when compared to itsSAFE, but it is still higher than the CPU of EcoTrec. In particular, BSTS reduces the CPU time by 99% when compared to itsSAFE. Eventually, it is important to notice that even with CPU time higher than EcoTrec the average CPU time for BSTS considering the worst case scenario (e.g.,
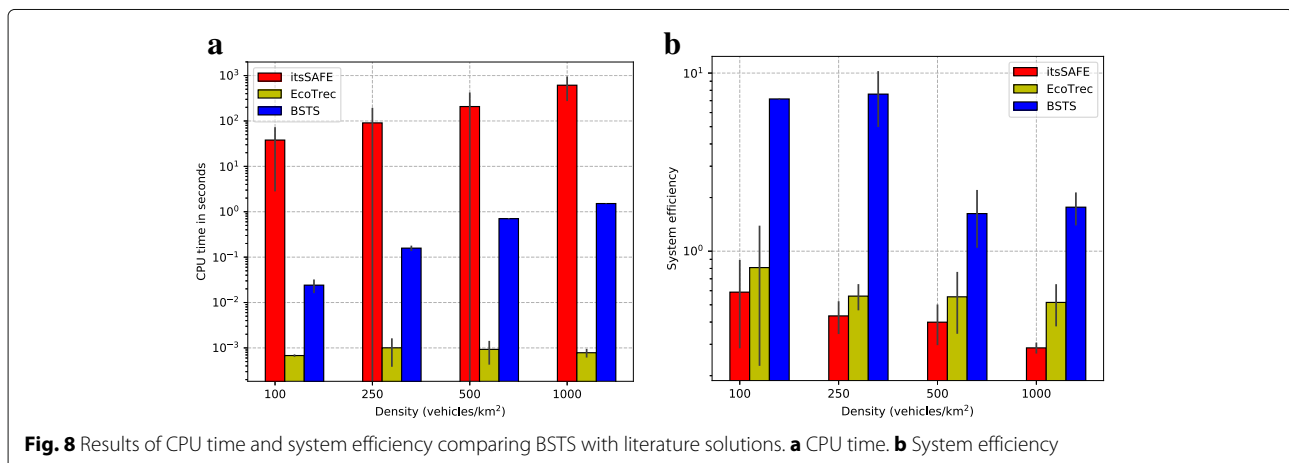
1000 vehicles/km^2) is lower than 1.5 s, which shows to be a suitable solution to enable real-time re-routing.

Figure 8b shows the average system efficiency for each density of vehicles. Thanks to the efficient traffic reporting mechanism and also to the cooperative re-routing algorithm employed by BSTS, it reduces the overhead while being able to deal with traffic efficiency issues properly. In this way, BSTS provides substantial improvements in the system efficiency when compared to itsSAFE and EcoTrec, which have shown to be costly solutions. In our scenario, BSTS is able to improve the system efficiency by at least 6 and 4 times compared to itsSAFE and EcoTrec, respectively.

4.5 Evaluation of the trade-off between traffic efficiency and safety risk

To evaluate the trade-off between the traffic efficiency and the safety risk, we compared the performance of BSTS with the following literature solutions: itsSAFE [9], EcoTrec [15], and SafePaths [16] concerning the following metrics:

- **Travel time** is the time spent for each vehicle to complete its route P from origin s and destination t. This time is obtained by computing the difference between the time that the vehicle enters into the simulation with respect to the time that it left. Also, this metric measures the overall traffic efficiency since long travel time can be a strong indication of traffic congestion
- **Safety risk** is the total safety risk of each vehicle to complete its route P from origin s and destination t. The safety risk is computed based on the safety risk of each road of its route (e.g., $\sum_{uv \in P} r_u v$). This metric evaluates how dangerous a route is (considering the public safety data used). Thus, high values of safety risks mean that a vehicle potentially had passed by one or more dangerous roads.

Fig. 8 Results of CPU time and system efficiency comparing BSTS with literature solutions. **a** CPU time. **b** System efficiency

Fig. 9 Traffic efficiency and safety risk results: comparing VNS, itsSAFE, SafePaths, EcoTrec and BSTS. **a** Average travel time. **b** Average route safety risk
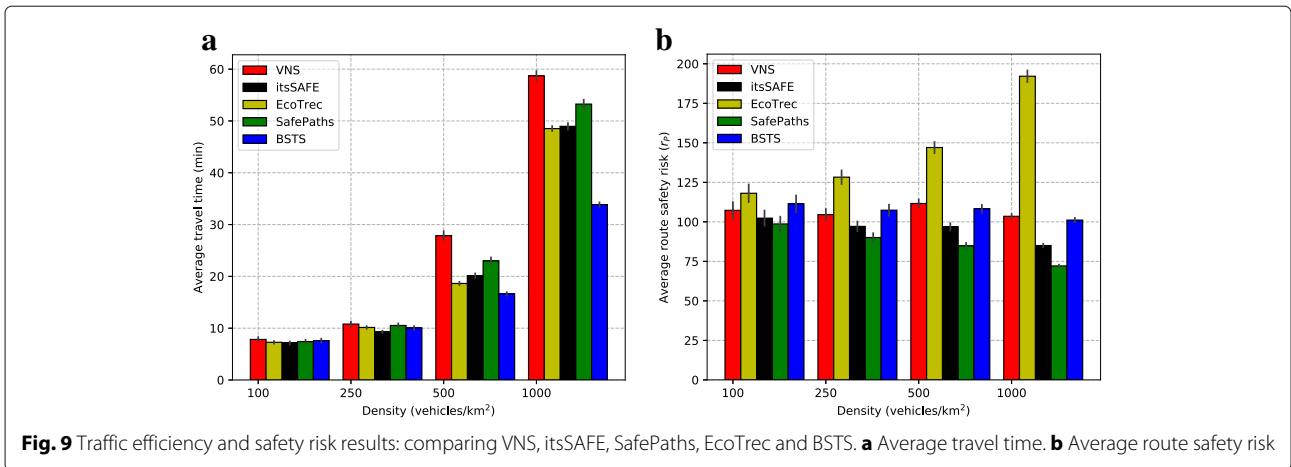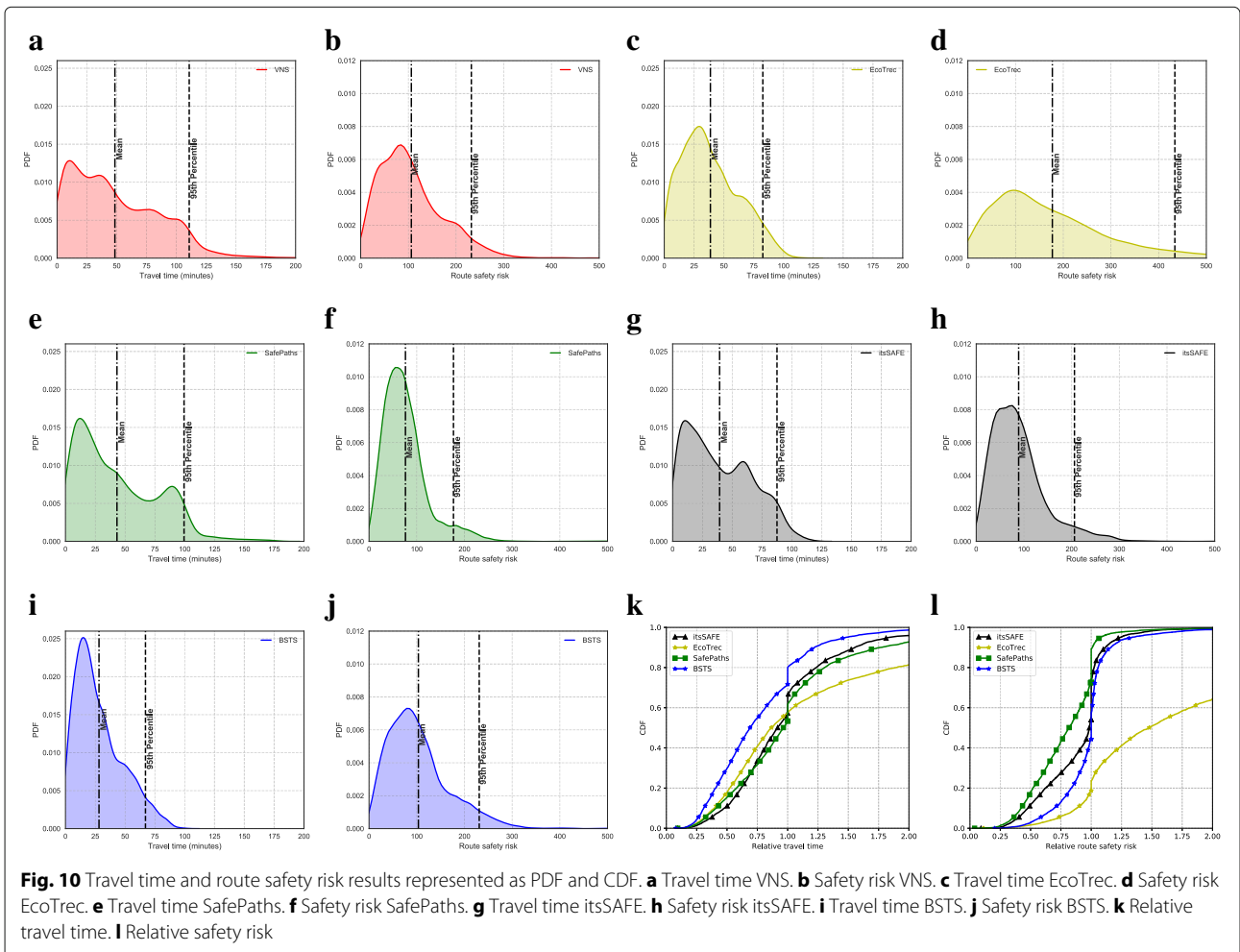
Figure 9 shows the results in function of the density of the vehicles for the average travel time (Fig. 9a) and the average route safety risk (Fig. 9b) for each solution. In addition, Fig. 10 shows the travel time and safety risk as a probability density function (PDF) and also as a

cumulative distribution function (CDF) of all densities concerning each solution.

The VNS results show the common scenario characteristics, since it does not apply any re-routing mechanism to improve neither mobility nor safety. In this way, we can

Fig. 10 Travel time and route safety risk results represented as PDF and CDF. **a** Travel time VNS. **b** Safety risk VNS. **c** Travel time EcoTrec. **d** Safety risk EcoTrec. **e** Travel time SafePaths. **f** Safety risk SafePaths. **g** Travel time itsSAFE. **h** Safety risk itsSAFE. **i** Travel time BSTS. **j** Safety risk BSTS. **k** Relative travel time. **l** Relative safety risk

observe that as the density of the vehicles/km^2 increases the average travel time goes up, varying from 10 min with 100 vehicles/km^2 to approximately 60 min with 1000 vehicles/km^2 (see Fig. 9a). Regarding the safety risk, VNS keeps the risk around 100 for all densities (see Fig. 9b). Analyzing the PDF for the travel time and safety risk of VNS, we have a better understanding of their behavior. As it is shown in Fig. 10a and b, 95% of the vehicles finish their trip with less than 110 min and a risk lower than 250.

On the other hand, EcoTrec re-routes vehicles using the shortest path algorithm based on the travel time on the roads, consequently computing faster routes to each vehicle. Therefore, EcoTrec reduces the average travel time by up to 18% when compared to the original mobility with density of 1000 vehicles/km^2(see Fig. 9a). However, EcoTrec presents two limitations: *(i)* lack of knowledge about risky areas, which consequently increases the average route safety risk by approximately 95%, since it does not care about safety issues while computing the fastest route to each vehicle (see Fig. 9b); and *(ii)* deterministic re-routing algorithm that decreases its overall efficiency as the scenario becomes denser. This is because many vehicles (with similar origin and destination) will compute the same route to avoid traffic congestion which creates different congestion spots. The PDF of travel time and safety risk shows that 95% of the vehicles ends their trip with less than 80 min (see Fig. 10c) with risk lower than 430, which is very high risk compared to VNS (see Fig. 10d).

SafePaths, in turn, re-routes the vehicles based on the safest routes, but as it does not have any knowledge about traffic conditions, it is not able to manage the traffic effectively. In this way, SafePaths minimizes the average route safety risk by approximately 30% and 60% compared to VNS and EcoTrec with a density of 1000 vehicles/km^2 (see Fig. 9b). However, considering the traffic efficiency SafePaths provides the average travel time close to the original scenario mobility because while improving safety without knowing the current traffic conditions on the roads potentially creates different congestion spots, consequently reducing the overall traffic efficiency. The rise of different congestion spots is shown in Fig. 10e in which the 95*th* percentile is only 10% lower when compared to VNS and 25% higher when compared to EcoTrec. However, Fig. 10f shows that the majority of the vehicles has a route safety risk lower than VNS and EcoTrec.

itsSAFE does not know only the current traffic conditions but also the safety risk on the roads, so it uses a bi-objective re-routing algorithm to provide a good trade-off between traffic efficiency and safety risk. In this way, it can decrease the average safety risk by about 20% while decreasing the average travel time by approximately 18%. Yet, like EcoTrec, itsSAFE also implements a deterministic

re-routing algorithm, which reduces its performance as the scenario becomes denser. Naturally, it also creates different congestion spots (see the 95 percentile in Fig. 10g similar to the one presented by the SafePaths solution). On the other hand, it improves the safety for the majority of the vehicles (see that the density distribution in Fig. 10h is close to the distribution presented by SafePaths).

BSTS uses a non-deterministic multi-objective re-routing algorithm that balances the traffic flow over the set of routes available in the Pareto set (e.g., set that optimizes both safety and traffic efficiency). Hence, it presents a better traffic efficiency than literature solutions, which gives a reduction in the average travel time of about 50% when compared to VNS. Moreover, BSTS also shows an improvement in the average route safety risk of 15%. The effectiveness of traffic balancing employed by BSTS is shown in the travel time distribution (Fig. 10i), where 95% of the vehicles finish their trip earlier than 60 min (e.g., the average travel time of VNS). These results also show that different from the other solutions, BSTS does not create different congestion spots, consequently providing a smooth traffic flow.

Finally, Fig. 10k and l show the CDF for the results as a relative values in respect to the VNS (e.g., the ratio between the metrics without any re-routing over the results of each re-routing solution). In summary, these results highlight the better performance of BSTS since it is able to reduce the travel time for 80% of the vehicles compared to VNS (see values lower than 1 in Fig. 10k) while improving the safety for 60% of the vehicles compared to VNS (see values lower than 1 in Fig. 10l). Another important aspect to observe is that the number of impaired vehicles in BSTS (e.g., vehicles that an increase on their travel time or their safety risk) is lower than 10% in both metrics (see values greater than 1 in Fig. 10k and l). However, when analyzing the other solutions, we can see that despite of the improvement in the travel time for 60% of the vehicles, they potentially impair 40% of them. In particular, such impairment on the traffic efficiency can grow up to twice more than the expected travel time (e.g., travel time presented by VNS) and to more than 5 times when considering the safety risk (see EcoTrec values on Fig. 10l).

With these results we can conclude that: *(i)* considering only traffic conditions is not enough to improve the safety of drivers and passengers as well as considering just safety issues does not provide efficient traffic mobility; *(i)* using deterministic approaches for dealing with multi-objective routing problems do not provide the desired efficiency in terms of traffic management, since many vehicles can take the same route which will create bottlenecks in the transportation infrastructure (e.g., different congestion spots) as the scenario becomes denser.

4.6 BSTS: k fastest routes vs k safest routes

Considering that vehicles can filter the Pareto set to use only the k fastest or safest routes during the route selection. In this section, we evaluate how this customization can impact on the performance of BSTS. To this evaluation, we considered only the density of 1000 vehicles/km^2 with $k \in [1, 3, 7]$ based on the following metrics: *(i)* travel time; *(ii)* route safety risk; and *(iii)* re-routing efficiency, which is that ratio between the safety risks and travel time. The re-routing efficiency represents the trade-off of each customization (e.g., the higher the value the better the re-routing efficiency)

Figure 11 shows the results for the assessed metrics considering the BSTS with the k fastest routes and with the k safest routes from the Pareto set $\mathcal{P}$. As expected for $k = 1$, BSTS does not perform the traffic flow balancing, consequently, it creates different congestion spots (like EcoTrec, SafePaths and itsSAFE in the previous evaluation). In other words, by selecting the fastest and the safest route from the Pareto set, BSTS provides performance very similar to EcoTrec and SafePaths, since these are the shortest path solutions considering traffic efficiency and safety, respectively. Therefore, with $k = 1$ BSTS has the worst average travel time for the safest and fastest routes with an average travel time of approximately 50 min. However, regarding the safety risk, with $k = 1$, the fastest approach provides the worst results while the safest one provides the best results.

Yet, with $k = 3$ and $k = 7$ BSTS is able to perform traffic balancing for both approaches (e.g., the k safest and the k fastest). Thus, considering the k safest routes of the Pareto set, BSTS reduces the average travel time in approximately 5 and 10 min with $k = 3$ and $k = 7$ in comparison with $k = 1$, respectively. On the other hand, it increases the average route safety risk, but such increase is only 5% and 7% higher than the average route safety risk of $k = 1$. Regarding, the fastest k routes of the Pareto set with $k = 3$ and $k = 4$, BSTS reduces the average travel time in 12
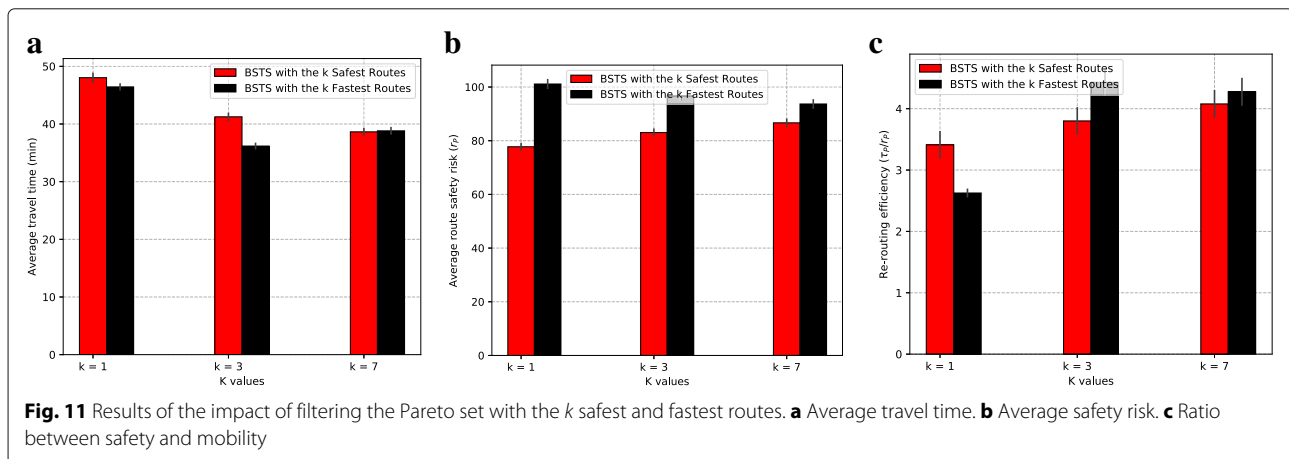
and 6 min when compared with $k = 1$, respectively. In addition, it also reduces the average safety risk since it has more safe options to choose from.

Finally, the effectiveness can be assessed by analyzing the re-routing efficiency of both approaches. Therefore, as it is shown in Fig. 11c the k fastest approach with $k = 3$ provides a better trade-off considering mobility and safety with the highest re-routing efficiency considering the metric $\frac{t_P}{r_P}$. However, considering only the k safest approach, the performance of $k = 7$ provides better effectiveness. This behavior is the result of the feasible routes available in each set (e.g., k fastest and k safest routes). In this way, the k fastest routes approach with $k = 7$ can include slower routes in the resulting set to choose the route of each vehicle. Meanwhile, the k fastest approach with $k = 7$ can include faster routes to be selected from the set of feasible routes.

With these results we can conclude that filtering the set of feasible routes can improve even more the performance of the BSTS. However, it needs to be tuned accordingly to each metric, since the k fastest and the k fastest routes can have different values for k to achieve the best performance. Eventually, it is important to stress that the re-routing algorithm employed by BSTS can also work with different metrics such as weather conditions, greenhouse emissions, pavement quality, and etc.

5 Conclusion

This work proposed BSTS, an ITS for vehicular traffic re-routing that considers both traffic conditions and public safety issues while computing alternative routes using a non-deterministic multi-objective re-routing algorithm. Also, BSTS implements an efficient traffic reporting protocol to minimize network contentions and improve the system scalability. Simulation results have shown that when compared to literature solutions for dealing with traffic and safety issues, BSTS presents a better trade-off between safety and traffic efficiency. It also minimizes the

Fig. 11 Results of the impact of filtering the Pareto set with the k safest and fastest routes. **a** Average travel time. **b** Average safety risk. **c** Ratio between safety and mobility

problem of creating different congestion spots, which is a limitation of deterministic approaches.

Specifically, BSTS reduces the average travel time up to 50% and improves the average safety risk by about 15%, when compared to original traffic mobility. However, considering individual travels, BSTS can reduce the travel time for 80% of the vehicles while improving the safety risk of 60% of them. It is worth noticing that we use the traffic efficiency and safety risk as a use case for BSTS, but the re-routing algorithm implemented in this paper can also work with other metrics that can impact in a route planning decision such as distance, greenhouse emissions, pavement quality, scenery, and etc.

As future work, we intend to investigate alternative approaches to define safety risks and incorporate social media data into the system to provide a more dynamic safety sensing. Moreover, we also plan to use machine learning techniques to predict future safety risk to be considered in the re-routing algorithm. Finally, a more complex decision making based on drivers' preference will be implemented to enable drivers to plan a route according to their personal preferences.

Abbreviations
5G: Fifth generation networks; BSTS: Better safe than sorry; CDF: Cumulative distribution function; DP: Dynamic programming; HCM: Highway capacity manual; ITS: Intelligent transportation systems; LOS: Level of service; PDF: Probability density function; OBU: On-board units; RCSP: Resource constrained shortest path; RSU: Roadside units; V2I: Vehicle-to-infrastructure; V2V: Vehicle-to-vehicle; V2X: Vehicle-to-everything; VANETs: Vehicular ad hoc networks; VNS: Vehicular navigation systems

Authors' contributions
All authors contributed equally. All authors read and approved the final manuscript.

Availability of data and materials
The datasets used and/or analyzed during the current study were provided by the Strategic Intelligence company and are available on reasonable request.

Availability of data and materials
The datasets used and/or analyzed during the current study were provided by the Strategic Intelligence company and are available on reasonable request.

Competing interests
The authors declare that they have no competing interests.

Author details
[1]Institute of Computing, University of Campinas, Campinas, São Paulo, Brazil. [2]Computer Science, University of Bern, Bern, Switzerland. [3]São Paulo State University (UNESP), Bauru, São Paulo, Brazil.

References
1. Amirgholy M, Golshani N, Schneider C, Gonzales EJ, Gao HO. An advanced traveler navigation system adapted to route choice preferences of the individual users. Int J Transp Sci Technol. 2017;6(4): 240–54. https://doi.org/https://doi.org/10.1016/j.ijtst.2017.10.001. http://www.sciencedirect.com/science/article/pii/S2046043017300345. Special Issue on Urban Spatiotemporal Behavior and Network Assignment.
2. Araujo GB, Queiroz MM, Duarte-Figueiredo FLP, Tostes AIJ, Loureiro AAF. CARTIM: A proposal toward identification and minimization of vehicular traffic congestion for VANET. In: Computers and Communication (ISCC), 2014 IEEE Symposium on; 2014. p. 1–6. https://doi.org/10.1109/ISCC.2014.6912491.
3. Behrisch M, Bieker L, Erdmann J, Krajzewicz D. SUMO - Simulation of Urban MObility: An Overview. In: International Conference on Advances in System Simulation (SIMUL '11). ThinkMind; 2011. p. 63–68. http://www.think mind.org/index.php?view=article&articleid=simul_2011_3_40_50150.
4. Board TR. Hcm 2010 - highway capacity manual. In: HCM 2010 - Highway capacity manual. National Research Council. Transportation Research Board of the National Academies; 2010. https://www.hcm2010.org/.
5. Botega LC, Oliveira ACM, Perira VA, Saran JF, Villas LA, Araujo RB. Quality-Aware Human-Driven Information Fusion Model. In: 20th International Conference on Information Fusion. IEEE; 2017. https://doi.org/10.23919/ICIF.2017.8009851.
6. Campolo C, Molinaro A, Iera A, Menichella F. 5g network slicing for vehicle-to-everything services. IEEE Wirel Commun. 2017;24(6):38–45. https://doi.org/10.1109/MWC.2017.1600408.
7. Cable News Network (CNN). Waze app directions take woman to wrong Brazil address, where she is killed. 2015. https://fox40.com/2015/10/06/waze-app-directions-take-woman-to-wrong-brazil-address-where-she-is-killed/.
8. de Souza AM, Botega LC, Garcia IC, Villas LA. Por aqui é mais seguro: Melhorando a mobilidade e a segurança nas vias urbanas. Anais do Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC). 2018;36:. http://portaldeconteudo.sbc.org.br/index.php/sbrc/article/view/2475.
9. de Souza AM, Pedrosa LLC, Botega LC, Villas L. Itssafe: An intelligent transportation system for improving safety and traffic efficiency. In: 2018 IEEE 87th Vehicular Technology Conference (VTC Spring); 2018. p. 1–7. https://doi.org/10.1109/VTCSpring.2018.8417760.
10. de Souza AM, Villas LA. A Fully-distributed Traffic Management System to Improve the Overall Traffic Efficiency. In: Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems, MSWiM '16. New York: ACM; 2016. p. 19–26. https://doi.org/10.1145/2988287.2989167.
11. de Souza AM, Yokoyama RS, Boukerche A, Maia G, Cerqueira E, Loureiro AAF, Villas LA. ICARUS: improvement of traffic condition through an alerting and re-routing system. Comput Netw. 2016;110:118–132. https://doi.org/http://dx.doi.org/10.1016/j.comnet.2016.09.011.
12. Djahel S, Doolan R, Muntean G-M, Murphy J. A communications-oriented perspective on traffic management systems for smart cities: Challenges and innovative approaches. IEEE Commun Surv Tutor. 2015;17(1):125–51.
13. Dong P, Zheng T, Yu S, Zhang H, Yan X. Enhancing vehicular communication using 5g-enabled smart collaborative networking. IEEE Wirel Commun. 2017;24(6):72–79. https://doi.org/10.1109/MWC.2017.1600375.
14. Doolan R, Muntean G-M. VANET-Enabled Eco-Friendly Road Characteristics-Aware Routing for Vehicular Traffic. In: Vehicular Technology Conference (VTC Spring), 2013 IEEE 77th; 2013. p. 1–5. https://doi.org/10.1109/VTCSpring.2013.6692807.
15. Doolan R, Muntean GM. EcoTrec: A Novel VANET-Based Approach to Reducing Vehicle Emissions. IEEE Trans Intell Transp Syst. 2017;18(3): 608–620. https://doi.org/10.1109/TITS.2016.2585925.
16. Galbrun E, Pelechrinis K, Terzi E. Urban navigation beyond shortest route: The case of safe paths. Inf Syst. 2016;57:160–171. https://doi.org/10.1016/j.is.2015.10.005.
17. Garip MT, Gursoy ME, Reiher P, Gerla M. Scalable reactive vehicle-to-vehicle congestion avoidance mechanism. In: IEEE Consumer Communications and Networking Conference (CCNC '15). IEEE; 2015. p. 943–948. https://doi.org/10.1109/CCNC.2015.7158103.
18. Joksch HC. The shortest route problem with constraints. J Math Anal Appl. 1966;14(2):191–197. http://dx.doi.org/10.1016/0022-247X(66)90020-5.

19. Kim S, Liu W. Cooperative autonomous driving: A mirror neuron inspired intention awareness and cooperative perception approach. IEEE Intell Transp Syst Mag. 2016;8(3):23–32. https://doi.org/10.1109/MITS.2016.2573339.

20. Kirkpatrick S, Gelatt CD, Vecchi MP, Others. Optimization by simmulated annealing. Science. 1983;220(4598):671–80.

21. Liu J, Wan J, Jia D, Zeng B, Li D, Hsu C, Chen H. High-efficiency urban traffic management in context-aware computing and 5g communication. IEEE Commun Mag. 2017;55(1):34–40. https://doi.org/10.1109/MCOM.2017.1600371CM.

22. Marler R, Arora J. Survey of multi-objective optimization methods for engineering. Struct Multidiscip Optim. 2004;26(6):369–95. https://doi.org/10.1007/s00158-003-0368-6.

23. Marler RT, Arora JS. The weighted sum method for multi-objective optimization: new insights. Struct Multidiscip Optim. 2010;41(6):853–62. https://doi.org/10.1007/s00158-009-0460-7.

24. OMNET++". OMNeT++ Network Simulation Framework. 2016. http://omnetpp.org/.

25. Mailonline. Dash cam captures the terrifying moment Waze smartphone app directs a driver into a GUNFIGHT in Boston; 2016. https://www.dailymail.co.uk/news/article-3717925/amp/Dash-cam-captures-terrifying-moment-Waze-smartphone-app-directs-driver-GUNFIGHT-Boston.html.

26. Pan J, Khan MA, Sandu Popa I, Zeitouni K, Borcea C. Proactive vehicle re-routing strategies for congestion avoidance. IEEE; 2012. p. 265–72. https://doi.org/10.1109/DCOSS.2012.29.

27. Pan J, Popa IS, Borcea C. Divert: A distributed vehicular traffic re-routing system for congestion avoidance. IEEE Trans Mob Comput. 2017;16(1):58–72. https://doi.org/10.1109/TMC.2016.2538226.

28. Papaleondiou LG, Dikaiakos MD. Trafficmodeler: A graphical tool for programming microscopic traffic simulators through high-level abstractions. In: VTC Spring 2009 - IEEE 69th Vehicular Technology Conference; 2009. p. 1–5. https://doi.org/10.1109/VETECS.2009.5073891.

29. Peng Z, Gao S, Li Z, Xiao B, Qian Y. Vehicle Safety Improvement through Deep Learning and Mobile Sensing. IEEE Netw. 2018;32(4):28–33. https://doi.org/10.1109/MNET.2018.1700389.

30. Shah S, Bao F, Lu C-T, Chen I-R. CROWDSAFE: Crowd Sourcing of Crime Incidents and Safe Routing on Mobile Devices. In: Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, GIS '11. New York: ACM; 2011. p. 521–524. https://doi.org/10.1145/2093973.2094064.

31. Sommer C, German R, Dressler F. Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis. IEEE Trans Mob Comput. 2011;10(1):3–15.

32. Taha A, AbuAli N. Route planning considerations for autonomous vehicles. IEEE Commun Mag. 2018;56(10):78–84. https://doi.org/10.1109/MCOM.2018.1800135.

33. Wang P, Di B, Zhang H, Bian K, Song L. Cellular v2x communications in unlicensed spectrum: Harmonious coexistence with vanet in 5g systems. IEEE Trans Wirel Commun. 2018;17(8):5212–24.

34. Wang S, Djahel S, Zhang Z, McManis J. Next road rerouting: A multiagent system for mitigating unexpected urban traffic congestion. IEEE Trans Intell Transp Syst. 2016;17(10):2888–2899. https://doi.org/10.1109/TITS.2016.2531425.

35. Ye H, Liang L, Li GY, Kim J, Lu L, Wu M. Machine learning for vehicular networks: Recent advances and application examples. IEEE Veh Technol Mag. 2018;13(2):94–101. https://doi.org/10.1109/MVT.2018.2811185.

36. Younes MB, Boukerche A. Efficient traffic congestion detection protocol for next generation VANETs. In: Communications (ICC) 2013 IEEE International Conference on; 2013. p. 3764–8. https://doi.org/10.1109/ICC.2013.6655141.

37. Zhang W, Aung N, Dhelim S, Ai Y. DIFTOS: A Distributed Infrastructure-Free Traffic Optimization System Based on Vehicular Ad Hoc Networks for Urban Environments. SENSORS. 2018;18(8):. https://doi.org/10.3390/s18082567.

Publisher's Note