Software-Defined Traffic Light Preemption for Faster Emergency Medical Service Response in Smart Cities[1]

Nazila Bagheri[2]. Saleh Yousefi [1]. Gianluigi Ferrari[3]

Abstract

Proper management of rescue operations following an accident is one of the most fundamental challenges faced by today's smart cities. Taking advantage of vehicular communications, in this paper we propose novel mechanisms for the acceleration of rescue operations resulting in reducing fatalities in accidents. The proposed approach is based on a Software-Defined Traffic Light Preemption (SD-TLP) mechanism that enables Emergency Medical Vehicles (EMVs) to travel on a route with the minimum number of interruptions. The SD-TLP makes preemption decisions based on global knowledge of the traffic conditions in the city. We also propose mechanisms for the selection of the nearest emergency center and fast discharge of the route of EMVs. Furthermore, depending on the dynamic traffic conditions on the streets at the time of the accident, an appropriate rescue route is selected for the EMV before it begins to travel. The proposed approach is evaluated using the OMNET++ and SUMO tools over part of the city of Tabriz, Iran. The simulation results demonstrate that the method can reduce the average rescue time significantly. The proposed approach keeps the resulting disruption in city traffic acceptably low while trying to shorten the rescue time as much as possible.

Keywords: Smart City, Emergency Medical Vehicle (EMV), Rescue Time, Rescue Route, SDX, Lane-changing, Traffic Light Preemption.

---

[1] A shorter version of this work has been presented and appeared in the IEEE proceedings of 10th International Conference on Computer and Knowledge Engineering (ICCKE 2020).

✉ Saleh Yousefi (Corresponding Author)
s.yousefi@urmia.ac.ir

Nazila Bagheri
Nazila.baghri93@gmail.com

[2] Department of Electrical and Computer Engineering Urmia University, Urmia, Iran

Gianluigi Ferrari
gianluigi.ferrari@unipr.it

[3] Internet of Things (IoT) Lab, Department of Engineering and Architecture University of Parma, Parma, Italy

1- Introduction

Recently, the concept of “smart city” is attracting a significant interest. In modern smart cities, information and communication technologies are used more extensively than in the previous “less smart” cities. In particular, transportation systems constitute a significant part of modern life, especially in large smart cities. Given the unprecedented development of urban centers and the introduction of new services resulting from technological improvements, these systems increasingly need proper support for the management of resources and vehicles [1, 2].

One of the most important challenges in large cities is the minimization of mortality and financial loss due to accidents. According to the Global Road Safety Commission, accidents cause 1.3 million casualties and more than 50 million injuries every year. If quickly transferred to medical centers to undergo treatment, however, accident victims will quite likely recover. For this purpose, rescue vehicles need to be notified, arrive at the scenes, and transfer the injured to medical centers as quickly as possible [3]. Since rescue vehicles (such as fire trucks, ambulances, and police cars) have always been prioritized, when traveling along the routes and the intersections around the city, the strategies of *traffic light preemption* for Emergency Medical Vehicles (EMVs) are employed to ensure quick and nonstop travel [5]. The ultimate goal of all traffic light preemption strategies is to reduce the rescue time to a value lower than a standard (desired) threshold. In this paper, the time it takes until the EMV arrives at the accident scene from the emergency station will be referred to as *rescue time* and the route selected for the EMV will be denoted as *rescue route*.

Intelligent Transportation Systems (ITS) play an important role in coordinating traffic and ensuring safety, which helps solving many of the challenging issues in smart cities. Along with other technologies (such as machine vision, navigation, radar technologies), the Internet of Things (IoT) and vehicular communications are key technologies that play an important role in modern ITS. These two major enabling technologies are nowadays known as the Internet of Vehicles (IoV). The improvements in the context of information and communications technologies have made it possible to design and implement solutions to develop ITS and provide modern services such as promotion of safety, dissemination of useful information to drivers, and congestion avoidance [1, 2]. However, given today’s large urban centers and the presence of a large number of vehicles, traffic lights, Road Side Units (RSUs), the current architecture of communication technologies lacks flexibility and scalability in large-scale deployments. One of the most promising solutions to overcome this limitation relies on the concept of Software-Defined Everything (SDX), which is based on the separation of the data plane (executing agents) and control plane (decision-making agents). The main feature of SDX is centralized decision-making, which leads to better and more consistent decisions. Flexibility, planning capability, and centralized control can thus be added to the available infrastructures [4]. Most of the proposed traffic light preemption strategies currently focus on local intersections. On the basis of local detection of EMVs and next intersection clearance, the process often leads to unavoidable delays at the intersections. An alternative approach is used when the preemption system is applied to the entire route, from the initial location of the EMV to the accident scene. By interrupting regular traffic at intersections, the system provides a special green route for EMVs to let them travel through intersections with fewer stops. In this case, even though the rescue time is reduced, the regular urban traffic is affected negatively. Therefore, a tradeoff solution is needed to achieve the desired value of rescue time while keeping the negative effect on regular traffic as limited as possible [5].

In this paper, our objective is to minimize the intervention time of EMVs to an accident scene. For this purpose, in the first place, considering the current traffic conditions in the city at the time of the accident, an appropriate *rescue route* is specified for the EMV (before its departure). Then, during the travel toward the accident location, for further reduction of the rescue time, two mechanisms are applied: (1) a lane-changing mechanism, which is applied to vehicles ahead aiming at discharging the EMV route, and (2) a Software-Defined Traffic Light Preemption (SD-TLP) mechanism for enabling (ideally) nonstop EMV travel through intersections. In SD-TLP, a centralized controller is employed for the provision of a comprehensive view of and consistent access to all the traffic lights along the rescue route, together with a real-time knowledge of the traffic conditions in the intersecions on the route. This global view of routes and traffic lights leads to a shorter rescue time for EMVs and, at the same time, decreases the delay imposed on the regular city traffic. In other words, we aim at an approach that minimizes the desired rescue time, while the regular city traffic is disturbed as little as possible.

The remainder of the paper is organized as follows. Section 2 provides a review of the previous works. The proposed techniques and algorithms are described in Section 3. Section 4 focuses on performance evaluation. Discussions, conclusions, and suggestions for future works are carried out in Section 5.

2- Previous Works

Vehicular communications are considered one of the promising enabling technologies (along with other complementary technologies) for substantiating smarter ITS in smart cities. However, many challenges have to be faced. In [6], the authors investigate the main components of an ITS system and review the main advantages of Vehicle-To-Vehicle (V2V) communications in this area, describing the applications enabled by such communications, including: driver assistance; direction and route optimization; road safety; and traffic management. For instance, an alert mechanism for vehicular networks, referred to as DRIFT, is introduced in [7], which not only notifies accidents on highways, but also uses route switching mechanisms for other drivers to prevent greater congestion at accident scenes and to reduce travel time, fuel consumption, and pollution. However, the area of emergency management is a relatively new field. In the following, we review the literature on the role of vehicular communications in emergency management, in the following three categories.

2-1 Application of SDX to Vehicular Networks

In [4], the authors investigate the effect of Software-Defined Networking (SDN) on the flexibility of route planning in vehicular networks, by connecting a central controller to the vehicle through cellular communications. The simulation results demonstrate that the routing efficiency is higher with SDN than without. Moreover, they introduce a mechanism of failure recovery which maintains the packet delivery rate at an acceptable level in case of disconnection from the central controller. Another technique for emergency message transfer using SDN is proposed in [8]. On the basis of the global view provided to the SDN controller, the most appropriate RSU and route are selected for message transfer, and V2V message dissemination is carried out in regions not covered by any RSU. The work of [3] investigates the effect of the addition of fog computing to the vehicular SDN architecture. The new architecture is analyzed from three perspectives: network, system, and service. The simulation results demonstrate that the integration of fog computing and SDN may significantly reduce rescue time, particularly if a considerable number of vehicles available in the area accept the policies concerning emergency conditions.

2-2 Use of Traffic Light Control Techniques for Traffic Control

The authors of [9] provide an illustrative investigation of the effect of vehicle interaction with traffic lights. In the presence of such an interaction, a vehicle can receive traffic light information and continue to travel taking that information into account, so that it avoids congestion at the traffic lights.

Considering factors such as vehicle entry rate and maximum rate of vehicles’ travel through an intersection, in [10] plans for traffic light color switching are proposed, seeking to minimize average queue length and average waiting time. In [11], the congestion at each lane leading to an intersection is measured: a longer green light will be set if the measured traffic density exceeds a threshold. The work of [12] uses genetic algorithms to control smart traffic lights, which is applied to an intersection involving four two-way input roads with two sets of parameters; (i) queues of vehicles and pedestrians at the red lights and (ii) the numbers of vehicles and pedestrians traveling through green lights. In [13], the authors suggest a particle swarm algorithm, which seeks to control traffic lights over an area wider than the urban space. It begins with an initial population, involving particles’ locations and speeds. Based on the simulation results, the use of the proposed algorithm increases the number of vehicles arriving at their destinations and their average travel times. Moreover, in [14] the authors propose to reduce waiting time on all the streets leading to an intersection and to increase productivity. The presented algorithm guarantees that traffic is shared fairly at each intersection by the different flows.

2-3 Traffic Light Control for Prioritization of EMV

In [15], a number of existing methods for emergency vehicle preemption is reviewed in detail. The models are categorized into three distinct groups, namely: “routing-based,” “scheduling-based,” and “miscellaneous.” In [16], a radio modem is used by emergency vehicles to notify their presence. Upon reception of an emergency alert at an intersection, the traffic light corresponding to the lane of the sender remains green for a longer time and gets back to the normal state after the emergency vehicle crosses the intersection. In [17], a cost-effective system is designed and incorporated to detect the presence of an emergency vehicle at an intersection. Upon detection, the traffic light switches from normal operation of the traffic light to an emergency mode and, then, resumes normal operation after the emergency vehicle passes the intersection. A dynamic system for switching traffic lights is presented in [18], in combination with an emergency vehicle detection system to aid with the preemption performance and flow of traffic. The system uses a microcontroller equipped with an infrared sensor, as well as a microphone to monitor traffic and identify emergency vehicles. The work of [19] aims at presenting a dynamic preemption algorithm that seeks to establish a trade-off between two objectives: minimization of emergency vehicle travel time and minimization of delay for the other traffic. The authors of [5] present a method that enables control of one route with several intersections for an emergency vehicle to travel through. Besides the control mechanism, an appropriate route is specified in the paper for the emergency vehicle before its departure based on the traffic parameters. The work of [20] aims at making a systematic investigation and comparison of optimization methods for route specification and preemption for emergency vehicles with a focus on factors such as use of dynamic traffic data and minimization of the impact of preemption on other traffic. The authors of [21] focus on the transfer from the emergency vehicle signal preemption state back to the normal state after the emergency vehicle passes through the intersection. Given objectives such as increasing efficiency, providing fairness, and increasing the number of vehicles passing through the intersection at each cycle, a multi-objective transition optimization model is designed. The purpose of [22] is to present a dynamic scheduling algorithm for traffic lights that considers one or more emergency vehicles. The scheduling is based on a calculation of the longest green time for a traffic light Recently, [23] proposes a novel real-time traffic signal control strategy that is a “green wave’’ signal control strategy for emergency vehicles, ensuring the rapid driving of emergency vehicles by reducing road saturation in advance, providing emergency vehicles with signal preemption, and restoring the road network. The simulation was conducted utilizing the urban traffic simulator SUMO. In [24], integration of IoT and SDN is used to reduce the time it takes until an EMV arrives at the emergency area. In the proposal, emergency conditions arrive at a central controller. If a high-risk accident is identified, the scene will be marked and required traffic light preemption is performed. Furthermore, if possible, alternate routes are suggested to other vehicles to further smooth the travel of emergency vehicles. The purpose of [25] is to present an emergency traffic management application for a SDN controller. The application provides the emergency vehicle with a quick, efficient route, presents a purposeful information system for the reduction of congestion on the rescue route, and seeks to reduce rescue time by controlling the traffic lights along the route, with little effect on the surrounding traffic. In [26], a novel mechanism utilizing SDN is introduced which, while providing an acceptable resuce time, does not damage other traffic by managing the behaviour of the traffic lights encountered in the rescue route. [27] proposes an optimal control strategy for emergency/preventive vehicle priority (EVP) using edge computing and IoT sensors for smart cities. This research was conducted using a GPS-based IoT sensor that continues to send location information (LI) to an edge server. The edge server calculates the optimal timing based on the proposed control strategy algorithm and clears the emergency vehicles.

In most literature works, during the preemption process only the next intersection the EMV will go trough is controlled. However, in our approach, because of the central control system, a general overview of the scenario parameters (such as the location of subsequent traffic lights as well as the traffic between the next two intersections) is taken into consideration to provide a more optimized route to the EMV. The environment of interest in our investigation involves a route with more than ten consecutive traffic lights, with dynamic traffic conditions at the accident time. Our work also includes a lane-changing mechanism for further reduction of the traffic in the accident area. Furthermore, we propose a post-preemption strategy (for bringing the traffic lights back to the normal state) using a global view of the urban traffic and intersections aiming at keeping the disturbance effect of traffic light preemption strategies as low as possible.

3- Proposed Approach

Figure1 shows an overview of proposed scenario.

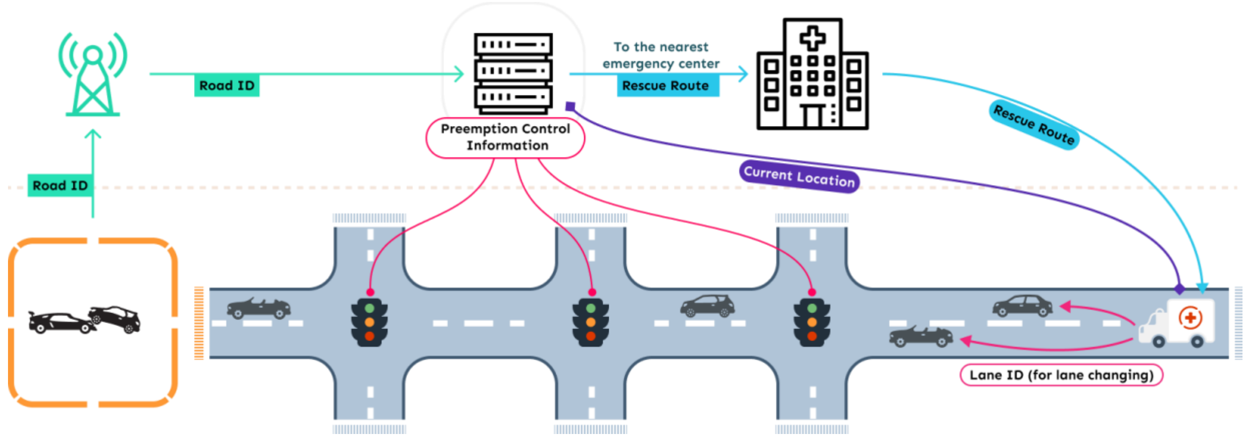

Figure 1. Overview of the Proposed Approach.

In our system model, we assume that there are some RSUs deployed all over the city. These RSUs can be IEEE 802.11p-enabled devices or LTE/5G base stations [28]. The RSUs are capable of sending/receiving messages to/from vehicles. Once an accident occurs, the involved vehicles send out their road segment identification to the closest RSU via Vehicle-To-Infrastracture (V2I) communication. We assume that this message arrives at a centralized SD traffic light controller by inter-RSU networking. The technology connecting RSUs is beyond the scope of this paper, but there are some candidate wired and wireless technologies (including Commercial Off-The-Shelf, COTS, solutions). Upon reception of information from the accident scene, the controller enables the proposed search mechanism for the closest emergency center, to which it sends the message after specifying the rescue route. The search and routing mechanism adopted in this step is based on Dijkstra’s algorithm [29]. We define the cost of each route based on its traffic density.

After receiving the accident scene identification, the emergency center dispatches an EMV through the controller-specified route, along which the EMV seeks to further reduce rescue time using *lane-changing* and *SD-TLP* mechanisms. In this section, we overview the proposed approach, which involves the following two mechanisms.

1- Detection of the closest emergency center to the accident scene and identification of the best (tentative) route from the emergency center to the accident scene.
2- Further reduction of the rescue time during the travel via:
    a- route discharge mechanism employing a lane-changing strategy for vehicles in front of the EMV;
    b- utilization of an SD-TLP mechanism along the route.

Each of the proposed mechanisms is detailed below, along with a description of its method of employment.

3-1 Selection of the Nearest Emergency Center and Identification of the Shortest Rescue Route

Identifying the nearest emergency center to the accident scene is the first efficient step, followed by determining the best rescue route. It should be noted that it is of great significance to use an appropriate measure of proximity (i.e., road segment cost), given the dynamicity of traffic conditions and its effect on road travel time. In the first component of our proposed approach, the controller searches for the closest emergency center and specifies the rescue route. Algorithm 1 (detailed below), which is Dijkstra's routing algorithm, is used to find the shortest path from the emergency center to the accident scene. The algorithm seeks, at each step, to determine the shortest path from the emergency center to the next intersection. In general, we denote $dist[v]$ as the distance between intersection *v* and the point of departure.

The algorithm begins at the emergency center and *dist* is obviously initialized to zero (lines 1 and 2). The set S of intersections with shortest distances is initiated to the empty set $\emptyset$. In lines 4 to 8, the value of *dist* is initially assumed to be infinity for all the city intersections; the value is updated as the algorithm progresses. The intersection *u* with the lowest value of *dist* is added, at each step, to set *S*, containing the intersections at the shortest distances from the point of departure, until the intersection associated with the accident scene is reached, as in lines 9 to 11. The value of *dist* is updated at each step as in line 13, where the cost of each path is calculated as $C_{uv} = L/S$, where L indicates the length of each road segment (dimension: [m]) and S is the average road segment speed (dimension: [m/s]).

As stated above, the algorithm is stopped from running when the intersection associated with the accident scene is reached.

Algorithm (1): Shortest Path Rescue Route

```
Inputs: EC, EC, wighted road city graph with $C_{uv}$
Output: The rescue route
//Symbols: dist: distance, EC: Emergency Center's location, AP: Accident Point's location, S: set of
    junctions with min cost $C_{uv}$: cost of the road from u to v

Function shortest path ($EC$):
        dist[$EC$] = 0                          //dist for source is zero
        S = $\emptyset$
    For each junction $v$ in city:
        If ($v \neq EC$)
            dist[$v$] = infinity
        end If
    end For
    While $u \neq AP$ :                         //$EMV$ did not reach to $AP$
        $u$ := junction not in S with min dist[$u$]
        add $u$ to $S$
        For each neighbor $v$ for $u$:
            dist[$v$] = min(dist[$v$], dist[$u$] + $C_{uv}$)     //$C\_uv = L/S$
        end For
    end While
end Function
```

Algorithm 1. Shortest Path Rescue Route.

Although the appropriate route is chosen for the EMV as it prepares to leave in order to guide it to the accident scene, the rescue route should also be controlled adaptively, given the dynamic nature of the traffic conditions and the potential for congestion at a specific part of the selected route. In the following, we propose two mechanisms aiming at this purpose.

3-2 Route Discharge via Lane-changing of Vehicles Ahead

Upon departure, the EMV can notify other vehicles of its presence utilizing V2V communications (e.g., using DSRC-enabled devices). Algorithm 2 (detailed below) relies on this mechanism. In lines 2 and 3 of Algorithm 2, the EMV sends the ID of the lane it is traveling to vehicles present on the rescue route. In lines 4 to 9, vehicles that receive the message will attempt to discharge it if they are in the same lane as the EMV.

This algorithm tries to reduce the congestion of road segments in front of EMVs as much as possible and, therefore, reduces (to some extent) the rescue time. However, in the case of severe congestion or with the availability of only one lane for each vehicle to travel, it is impossible to change lanes and discharge the route for the EMV. In such conditions, a more efficient solution is required to face congestion, particularly at intersections. The next proposed mechanism addresses the intersection and tries to reduce the congestion at the intersections for the EMV.

Algorithm (2) : Lane Changing Algorithm

```
//Symbols: EMV: Emergency Vehicle’s ID, AP: Accident Point’s location
// Each vehicle receiving the larm message form the EMV changes the lane if it is the same as the lane of EMV

Function Lane Changing():
  While EMV did not reach to AP :
    send(lane of EMV)                          //  emergency vehicle keeps sending its lane number
      For all receivers:
        If (lane = lane of the EMV)            //if the vehicle is in the emergency lane
          change lane();
        end If
      end For
  end While
end Function
```

Algorithm 2. Lane Changing Algorithm.

3-3 Software-Defined Traffic Light Preemption (SD-TLP) on the Rescue Route

A more efficient solution for the reduction of rescue time is to employ the technique of route preemption for the EMV, *i.e.,* its provision with an exclusive route to travel nonstop. In our proposed approach, we utilize the idea of Software-Defined Everything (SDX), according to which the controlling agent of all traffic lights (including those available along the rescue route) is moved to a software-defined central controller. This controller can configure traffic lights by any southbound API interface. The comprehensive global view of the traffic lights and traffic in their surrounding helps us to plan a more efficient preemption strategy. Algorithm 3 lists the steps of the proposed approach. The parameters used in Algorithm 3 are summarized in Table 1.

Table 1. Main parameters used in Algorithm 3.

Parameter	Definition
ET	Estimated time required for the discharge of the queue at the traffic light on the current road segment and appropriate time to begin preemption operation(seconds)
C	Number of vehicles present on the road segment
L	Average vehicle length (meters)
MG	Minimum Gap between vehicles (normal city traffic: 2.5 m, congested traffic: 1 m)
S	Current road segments' average speed (meters per second)
DD	Appropriate distance from the EMV to the traffic light for preemption to begin (meters)
EVS	Current EMV speed (meters per second)
r	A constant numerical value representing the delay in traffic light phase switching (3 seconds)

The proposed algorithm is composed of three parts. The *first part* involves a calculation of the appropriate time and location for the preemption operation to begin. For this purpose, the EMV needs to send its location information to the RSUs until it arrives at the accident scene, as indicated in lines 2 and 3. If its entry onto a new road segment is detected by the software-defined controller, the values of an estimated time for discharge of the current road segment and detection distance are calculated for the new road segment based on (2) and (3), as indicated in lines 5 and 6:

$$ET = \frac{C \cdot L + \left((C-1) \cdot MG\right)}{S} + r \tag{2}$$

$$DD = ET \cdot EVS. \tag{3}$$

The goal of (2) is to determine an estimate of the amount of time needed by the present road segment to be discharged at the moment of entry onto the road segment. Given ET, there will be no need for the traffic light to turn green on the verge of entry onto the road segment, which would waste green light time and impose extra traffic congestion at the intersections. The appropriate time for the control process to begin is when the EMV arrives at an appropriate distance from the traffic light, calculated in turn from (3) using the time obtained from (2) and the EMV speed.

After DD is calculated, the control operation begins once the EMV is as close as DD to the traffic light, as indicated in lines 13 to 20 of Algorithm 3. As a result, the red light time duration is reduced in the control (line 15), but green-light time duration is prolonged (line 18).

Algorithm (3): Software-defined Traffic Light Preemption (SD-TLP)

Inputs: *EMV's location, Traffic lights location in the rescue route, AP, C, L, MG, S, EMVS,*
Action: *The appropriate time and location for the preemption operation to begin, changing the traffic lights to green, changing the traffic lights to green*
//Symbols: EMV: Emergency Vehicle's ID, AP: Accident Point's location, TL: traffic light's ID, Dtls: The distance between two consecutive traffic lights, Dtreshold: threshold distance between two consecutive traffic light, DEMV: Distance of emergency vehicle to traffic light, TD: traffic density

```
1:  Function SDX-based Preemption (AP):
2:      While  EMV did not reach to AP :
3:          EMV send (road id)
4:              If it is a new road segment
5:                  compute ET using equation(2)
6:                  compute DD using equation(3)
7:                  For previous traffic lights:                  //traffic lights recovery mechanism
8:                      If (Dtls < Dtreshold)
9:                          Set second TL phase as first TL phase
10:                     end If
11:                 end For
12:             end If
13:             else If (DEMV <= DD)                              // if the EMV is within the detection distance
14:                     If (TL phase = "red")
15:                         reduce phase duration;
16:                     end If
17:                     If (TL phase = "yellow")
18:                         change phase to green;
19:                     end If
20:             end If
21:             For the next two lights:
22:                 If (TD > threshold)                           // if traffic density is high
23:                     If (TL phase = "red")
24:                         reduce phase duration;
25:                     end If
26:                     If (TL phase = "yellow")
27:                         change phase to green;
28:                     end If
29:                 end If
30:             end For
31:     end While
32: end Function
```

Algorithm 3. Software-defined Traffic Light Preemption (SD-TLP) Algorithm.

In the *second part* of the algorithm, involving lines 21 to 30, congestion is checked for the next two traffic lights on the path. If it exceeds a given threshold, set to 70% in the following, the control operation is applied to the latter traffic lights as well, which will help confront the EMV with lighter traffic after it travels through the first traffic light. The threshold value in this part means that more than 70% of the road segment is covered by vehicles. The threshold value equal to 70% has been empirically chosen based on trial and error in the simulation environment.

- If the threshold value is set higher than 70%, congestion might exceed a tolerable limit before the EMV enters the road segment. In this case, control would be necessary before the EMV enters and, upon its entry, we would be forced to extend the green phase of the traffic light, leading to longer stops for vehicles in the intersection.
- If the threshold value is set lower than 70%, there should be no need to change the traffic light phase. In fact, unnecessary manipulation of the traffic light would disrupt the normal traffic flow without any significant benefits.

It will also cause traffic lights close to each other to exhibit similar control, preventing a lack of coordination as the EMV travels through them. Basically, this is made possible by a central software-defined controller that is aware of both the traffic on the rescue route and the EMV route.

The *third part* of the algorithm (post-preemption approach), involving lines 7 to 11, which is related to coordination check and preemption recovery; refers to the time interval during which the EMV has completed the traffic light control operation for an intersection. If the EMV enters a new road segment after traveling through an intersection, as in line 4, the distances between the previous traffic lights on the travelled route are calculated in order for the coordination between those close to each other not to be disturbed. If the distance is below a given threshold, consecutive traffic lights are synchronized so that they will switch back to their normal state at the same time, preventing a disordered modification of the way in which vehicles travel the rescue route after the rescue operation has completed.

4- Performance Evaluation of the Proposed Approach

In this research, we use OMNET++[4] as a network simulator, SUMO[5] as a tool for representation of mobility around the city, and VEINS[6] as a tool for the establishment of a link between the network simulator and the mobility simulator [30, 31,32].

The urban location selected for implementation of the proposed approach is a district in the city of Tabriz, East Azerbaijan Province, Iran, a map of which has been extracted from the Open Street

---

[4] Objective Modular NETwork Testbed (OMNET).
[5] Simulation of Urban MObility (SUMO).
[6] VEhicles In Network Simulation (VEINS).

Map website [33]. Relevent information on the considered scenario and simulation parameters is summarized in Table 2.

Table 2. Main simulation parameters in the considered scenario.

Parameter	Value
OSM bound box	46.2760,38.0817,46.3045,38.0725
The map	Tabriz city
Vehicle transmission Power (to send/receive messages)	30 mW
RSU transmission Power	60 mW
Number of emergency centers	2
Number of RSUs	8
Number of SDN controller	1
Simulation time	1800 s

.

The extracted urban map is shown in Figure 2 along with the locations of the emergency centers.

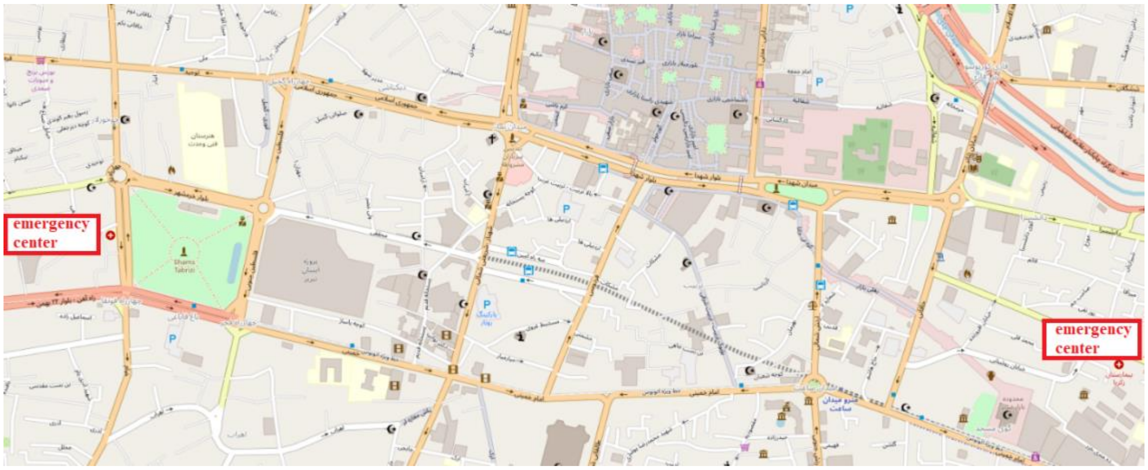

Figure 2. Portion of the city of Tabriz, Iran, in which our approach is simulated.

4-1 Performance Metrics

The purpose of this paper is to provide appropriate management of emergency conditions after the occurrence of accidents. In this regard, the first step is to reduce the time required by an EMV to arrive at the accident scene from the emergency center, *i.e.*, to reduce rescue time. The following three performance metrics are considered.

- **Average Rescue Time (ART)**

As previously defined, ART is the average time it takes an EMV to arrive at the accident scene from the emergency station (center).

- **Average Imposed Waiting Time (AIWT)**

Appropriate management is provided so that the mechanisms developed for decreasing the rescue time have the lowest possible impact on the normal city traffic. In other words, the delay imposed at intersections along the rescue route on vehicles traveling routes intersecting that of the EMV is kept reasonably low. Therefore, the second criterion involves the *average waiting time imposed on vehicles traveling on the routes that intersect that of the EMV.*

- **Average Waiting Time on Rescue Route (AWT)**

Due to the phase switching made to the traffic lights available along the route, the coordination between traffic lights close to each other may be disturbed once the rescue operation is completed: this will have a negative impact on the vehicles traveling along the rescue route, thereby increasing their delay. Therefore, the proposed approach should prevent disorder by considering a recovery mechanism. Thus, the third criterion involves the *average waiting time for the vehicles traveling on the same route as that of the EMV.*

4-2 Performance Evaluation

For assessment, we implement and consider the following six scenarios.

- **Normal State**

The normal, *i.e.* default, traffic state with regular traffic light functionality and no additional implemented mechanism.

- **Lane-changing**

Implementation of the mechanism proposed for lane-changing of vehicles ahead.

- **Full Preemption**

Implementation of a strategy where all the traffic lights along the route are green once the EMV begins to travel.

- **Local Preemption**

Implementation of the strategy of controlling traffic lights locally and independently.

- **SDN-based Preemption**

Implementation of the proposed traffic light control mechanism (SD-TLP).

- **SDN-based Preemption + Lane-changing**

Implementation of the SD-TLP along with the mechanism proposed for the lane-changing mechanism.

The accident occurs 300 seconds after the beginning of the simulation, considered as simulation warm-up time, during which the number of vehicles traveling in the city reaches a relevant (and steady) value. In order to ensure that the proposed approach performs effectively, we replicate the simulation for ten random days. We measure the performance metrics outline above (namely: AIT, AIWT, AWT) for different congestion modes, in each implementation, in different traffic conditions by adding the flow of vehicles to the rescue route with different periods besides the current traffic in the city.

We obtain rescue time in the Normal state, with no management mechanism of emergency conditions, and measure thw current traffic in the city, *i.e.*, the waiting times of other vehicles at the intersections along the rescue route. The performance is assessed for different numbers of vehicles traveling along the rescue route. In all performance graphs, the horizontal axis represents the average number of vehicles traveling the route ahead of the EMV from the time instant at which it begins to travel to the time instant it arrives at the accident scene. This number, denoted as N_v, is calculated for each road segment upon entry onto it, and the final numbers are obtained by averaging over ten days, each involving the flow of vehicles with different periods.

AIWT and AWT in the control scenarios are calculated after the additional period from the arrival of the EMV at the accident scene. In fact, the additional period is considered for examination of the effect of the control mechanisms on the performance of traffic lights after returning from emergency to stable conditions.

4-2-1 Evaluation of the Effect of the Lane-changing Mechanism

Figure 3 shows the ART obtained from the implementation of the lane-changing scenario, as compared to that in the Normal scenario.

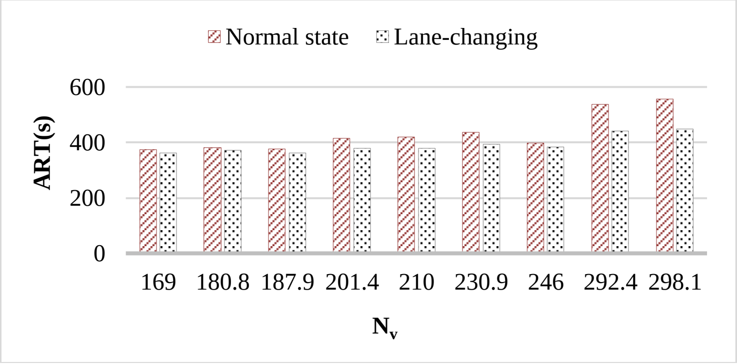

Figure 3. ART of the lane-changing scenario, as compared to the ART of the normal scenario.

On the basis of the obtained results, it can be concluded that the lane-changing mechanism reduces the rescue time to some extent, but the amount of improvement cannot exceed a certain level due to the presence of obstacles such as: red lights along the rescue route; single-lane routes; the impossibility of lane-changing because of congestion.

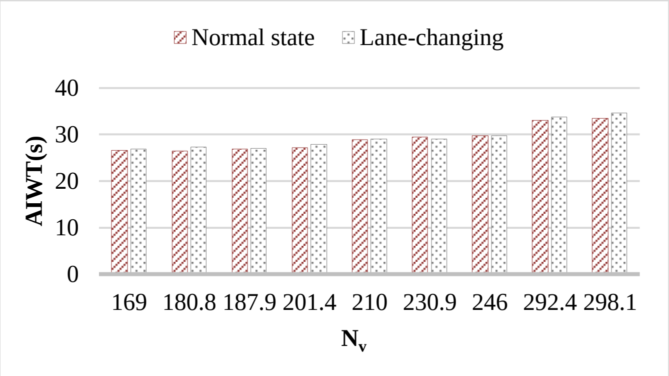

Figure 4. AIWT of the lane-changing scenario, as compared to the AIWT of the normal scenario.

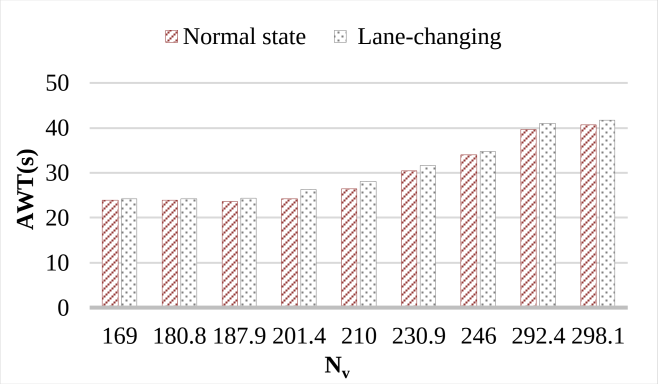

Figure 5. AWT of the lane-changing scenario, as compared to the AWT of the normal scenario.

Figure 4 and figure 5 show the AIWT and AWT along the rescue route, respectively, for the lane-changing method. As it is clear from the obtained results, the lane-changing mechanism fails to considerably affect the waiting times of other vehicles, since it applies no control operation to the traffic lights along the route.

For a considerable reduction of the rescue time, traffic light control techniques need to be used. As mentioned earlier, the shortest rescue time will be achieved if all the traffic lights along the route are green and the EMV travels the route nonstop (full preemption strategy).

4-2-2 Evaluation of the Effect of the Full Preemption Mechanism

The full preemption scenario can be considered the ideal scenario for the EMV. However, it affects the normal traffic of the city quite negatively. As a result, it cannot be used in practice. Nonetheless, we can use it as a benchmark for assessing other mechanisms in our approach. In Figure 6, we compare the ART in the normal condition (when no mechanism is in place) and with full preemption (ideal condition; when all of the traffic lights along the path are green).

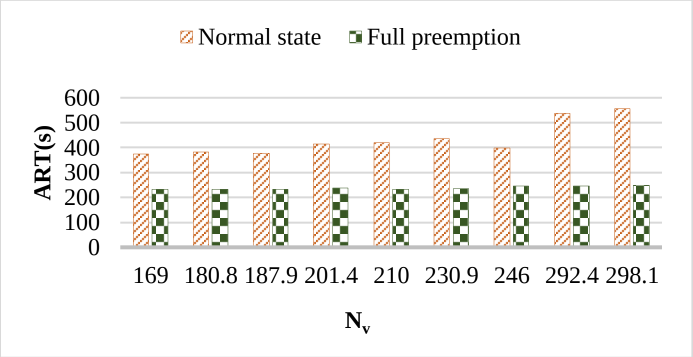

Figure 6. ART of the full preemption scenario, as compared to the ART of the normal scenario.

The observed results show that the ART often increases as does the average number of vehicles present on the rescue route. It should be noted that an increase in the number of vehicles does not necessarily increase the rescue time in some cases, where factors such as direction, travel lane, and vehicle routes are effective. If the full preemption method is applied, however, the average rescue time will be reduced considerably for any traffic volume, since all the traffic lights present along the rescue route will be green once the EMV begins to travel.

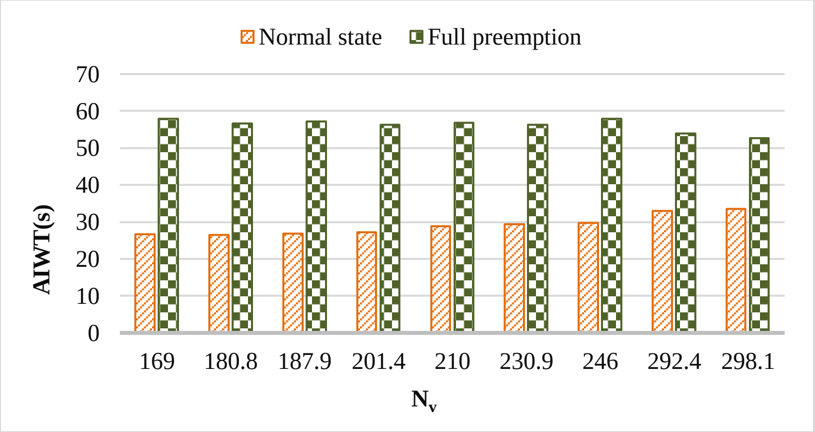

Figure 7. AIWT of the full preemption scenario, as compared to the ATWT of the normal scenario.

Figure 7 shows the AIWT criterion in both normal and full preemption conditions. From the results of Figure 6 and Figure 7, it can be concluded that the average delay imposed on intersecting vehicles at the intersections along the rescue route is not negligible in this scenario, although the best rescue time for the EMV is achieved. This is because all traffic lights along the rescue route switch to green once the EMV begins to travel: this is unnecessary and imposes a significant delay on vehicles traveling along routes intersecting that of the EMV.

Therefore, the appropriate approach is the one that can maintain AIWT close to that of the normal state and, at the same time, provide a low ART. In other words, rescue operation acceleration techniques should not influence public urban traffic to a great extent.

4-2-3 Evaluation of the Effect of the Local Preemption Mechanism

As noted earlier, an available solution in the literature that is proposed for the reduction of rescue time and its impact on public traffic is local traffic light preemption, where each traffic light is controlled independently of others. Figure 8 illustrates the steps required for the local preemption mechanism implemented in this paper. In the following figures, we evaluate the performance obtained with this mechanism with respect to Normal and full-preemption scenarios.

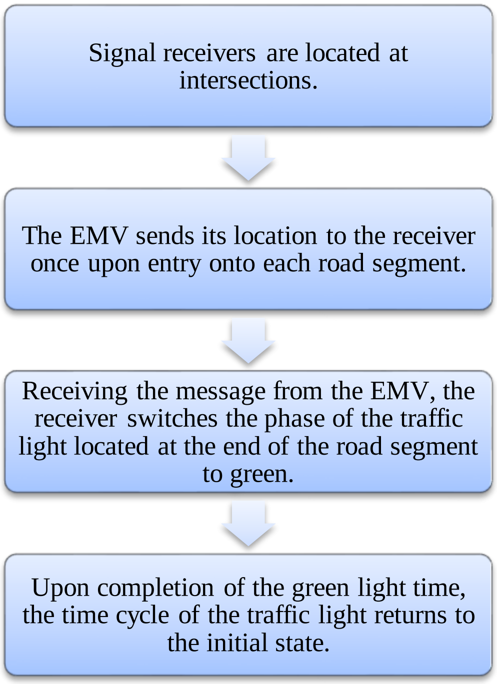

Figure 8. Local control mechanism of traffic lights on the rescue route.

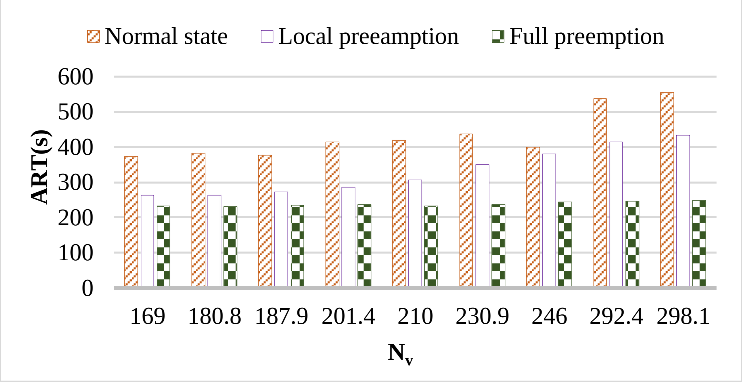

Figure 9. ART of the local preemption scenario, as compared to the ART of the previous scenarios.

Figure 9 shows that this method reduces rescue time to a desirable extent, particularly for light traffic conditions. The local preemption method leads to rescue times shorter than the normal conditions. However, the rescue time growth rate in the normal state and the local preemption are fairly similar. As traffic becomes heavier, the improvement effect of this policy is also limited. This can be explained by the following reasons.

1- There might be so much congestion at the road segments of a traffic light that the EMV fails to arrive within an appropriate distance from the traffic light to begin the preemption operation. In the local preemption scenario, there is no central controller and the EMV has to communicate with each traffic light directly.
2- At the local preemption, the EMV considers only the first immediate traffic light in its path and has no information about the next lights and the distance between them. In some cases, the traffic lights may be so close together that the behavior of a traffic light may affects the other. This effect causes inconsistencies in the traffic lights and disrupts traffic along the rescue route.

The proposed solution in our work (namely, SD-TLP) solves these limitations of local methods by relying on the general view of the central controller.

Figure 10 shows the AIWT, as a function of the average number of vehicles in the rescue route, in the following conditions: local preemption; normal; full preemption. The obtained results show that AIWT with local preemption is higher than that with full preemption.

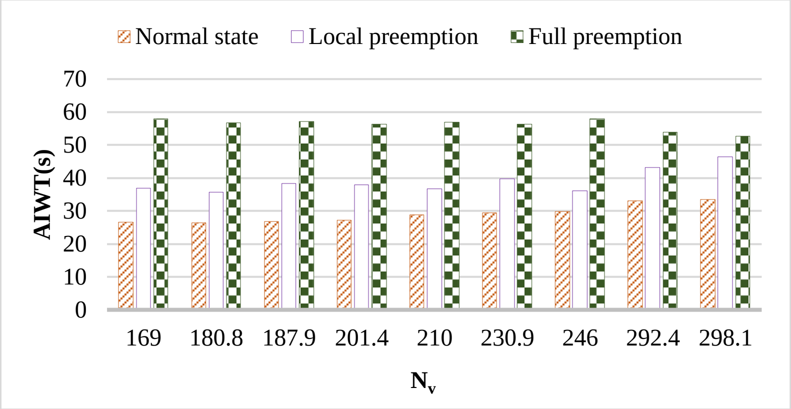

Figure 10. AIWT of the local preemption scenario, as compared to the AIWT of the previous scenarios.

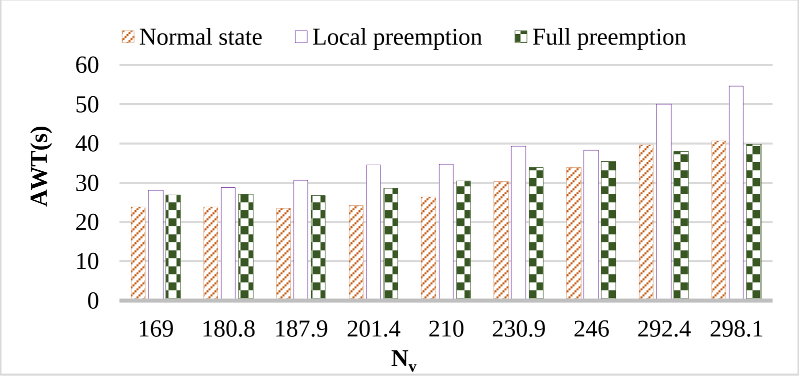

Figure 11. AWT of the local preemption scenario, as compared to the AWT of the previous scenarios.

Figure 11 shows the AWT for the Local preemption scenario. As can be observed from the results in in Figure 11, AWT increases as traffic congestion increases. In lighter traffic loads, the EMV can travel nonstop across the intersections; therefore, independent traffic light control performs acceptably. However, as traffic becomes heavier, it is more likely that the EMV travels through intersections with a longer delay (due to the above-mentioned reasons), which disturbs coordination among consequent traffic lights. This increases the AWT along the rescue route.

The advantages and disadvantages of the implemented scenarios are summarized in Table 3.

A desirable approach should guarantee the desired ART and keep AWT and AIWT as low as possible. The proposed lane-changing method, as can be inferred from Table 3, is insufficient on its own; if the number of vehicles is large and there is no traffic light control, congestion could build up behind red traffic lights at intersections, preventing vehicles from changing lanes. Therefore, this mechanism alone would not significantly reduce the ART for the entire route. As a result, it is necessary to use traffic light control mechanisms.

Implemented scenarios	Advantage	Disadvantage
Lane-changing	reduction of ART as compared to the Normal state scenario and induction of no disorder in current traffic in the city	Limited reduction of ART and failure to ensure the reduction in different traffic conditions
Full preemption	Provision of the lowest ART	Unacceptable increase of AIWT
Local preemption	Provision of balance in AIWT and proper ART and AWT in light traffic	Poor performance in reduction of ART in heavy traffic, along with interruption of coordination among traffic lights close to each other and consequent increase of AWT

Table 3. Advantages and disadvantages of the implemented scenarios.

In the following, in order to solve the disadvantages listed in Table 3, we will analyze the impact of the SD-TLP mechanism proposed in this work and evaluate its effectiveness.

4-2-4 Evaluation of the Effect of the SD-TLP Mechanism

The proposed SD-TLP mechanism aims at improving the drawbacks mentioned in Table 3.

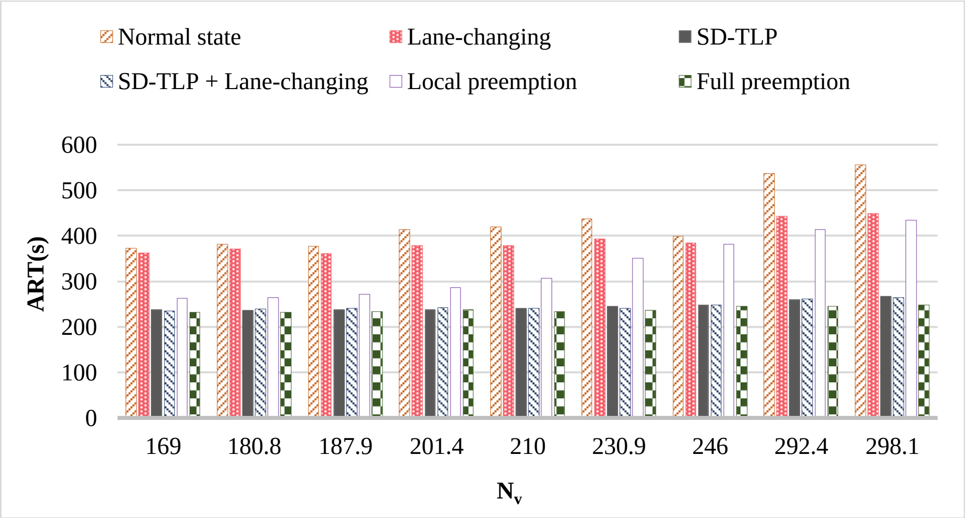

Figure 12. ART of the SD-TLP, as compared to the ART of the previous scenarios

Figure 12 shows the ART for the proposed approach as compared to the previous methods. As shown in the figure, the SD-TLP mechanism leads to a better ART with respect to other approaches. Indeed, SD-TLP can set the appropriate time and location to start traffic light control operations given the severity of congestion on the current road segment. The location of the EMV is continuously under control and there is no need for the EMV to be at a particular distance from the traffic light for triggering the preemption process. Furthermore, the rescue route is also controlled all the time to control congestions resulting from two subsequent traffic lights so that the EMV is not confronted with congestion at the subsequent intersections after traveling through the current one. This helps in keeping traffic light and guarantees near-optimal rescue time for any traffic volume.

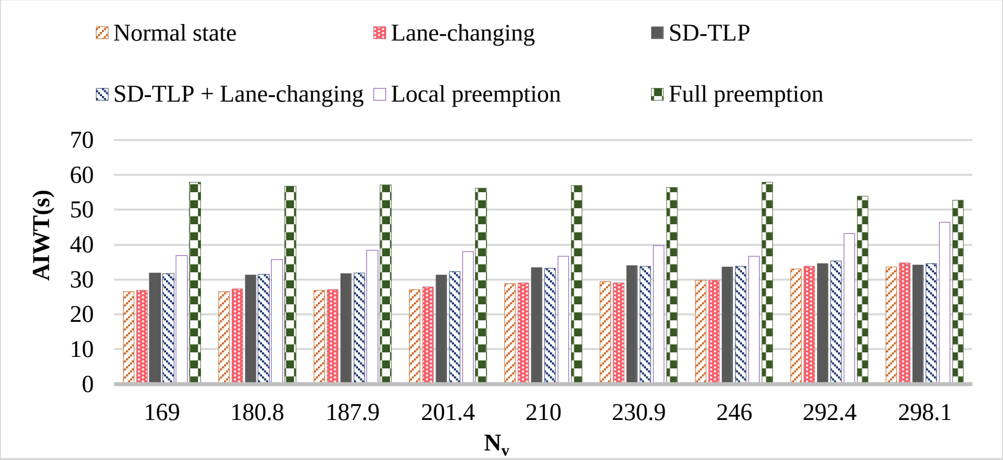

Figure 13. AIWT of the SD-TLP, as compared to the AIWT of the previous scenarios

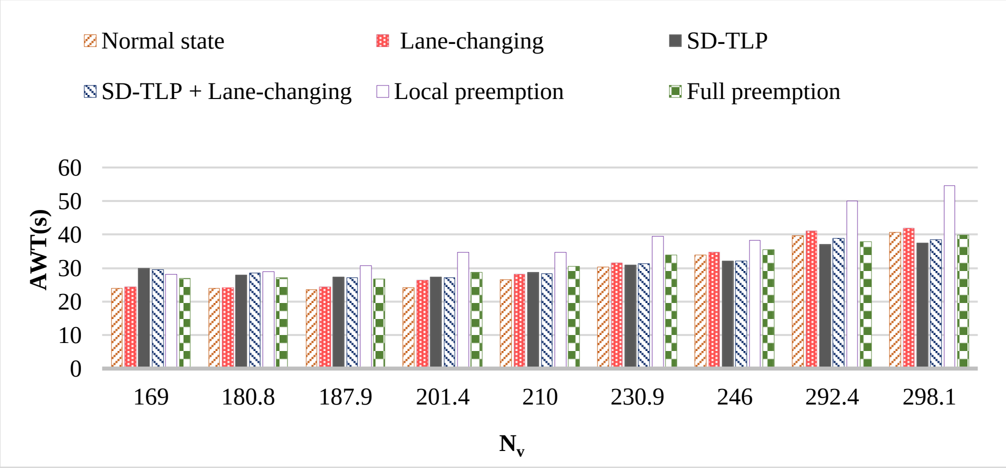

Figure 14. AWT of SD-TLP, as compared to the AWT of the previous scenarios

Figure 13 shows the AIWT for the proposed method as compared to the previous ones. Based on the obtained results, the proposed method can maintain AIWT at an acceptable level for the vehicles traveling on the routes that intersect the EMV route. SD-TLP performs better in terms of AIWT than local preemption, as the control operation begins at the appropriate time thanks to the global view provided by the centralized software-defined controller.

Finally, Figure 14 shows the AWT for the proposed method as compared to the previous methods. Based on the obtained results, in very light traffic, the local preemption approach performs slightly better than SD-TLP. This is due to the fact that, with local preemption, the traffic light phase is switched to green immediately upon entry onto each road segment (earlier than the phase change of lights in SD-TLP), and more vehicles on the same route of the EMV can pass through the intersection. In moderate and heavy traffic conditions, SD-TLP performs much better than the other control methods and can better manage the disorder caused by traffic lights, because of the extra care given in the process of returning encountered traffic lights to their previous state after the EMV has passed through.

Conclusions and future works

In this paper, taking advantage of the global view and knowledge of the city traffic that is provided by a centralized controller, we have first proposed a real-time efficient rescue route for the EMV and have applied SD-TLP preemption strategies for the EMV to travel with minimal interruption. Moreover, a lane-changing mechanism has also been proposed for the discharge of the rescue route. Light switching has been considered in order to minimize the disorder caused to regular urban traffic. On the basis of the obtained results, the proposed lane-changing mechanism can reduce rescue time to some extent, but the amount of reduction is not sufficient, as the EMV needs to stop frequently at intersections. Although it ensures optimal rescue time, the full preemption scenario induces the largest amount of disorder in regular traffic. Local preemption can maintain disorder at a more balanced level but fails to offer an acceptable rescue time due to the lack of information on the conditions of surrounding traffic lights. Finally, the SD-TLP method outperforms the others in terms of both rescue time and normal urban traffic disorder.

Interesting future research directions can be summarized as follows: 1) application of the proposed method to scenarios with larger numbers of EMVs of different types, such as fire engines and police cars, given different levels of priority; 2) use of different technologies such as LTE/5G for the communication between the controller and the network infrastructures and vehicles; 3) framing of the preemption scheduling problem as an optimization problem with current traffic data and looking for answers.

Acknowledgements

The work of Gianluigi Ferrari is partially supported by the OPEVA project that has received funding within the Key Digital Technologies Joint Undertaking (KDT JU) from the European

Union's Horizon Europe Programme and the National Authorities (France, Czechia, Italy, Portugal, Turkey, Switzerland), under grant agreement 101097267. Views and opinions expressed are however those of the author only and do not necessarily reflect those of the European Union or KDT JU. Neither the European Union nor the granting authority can be held responsible for them.

References

[1] R. Ipolito Meneguette, *Intelligent transport system in smart cities: aspects and challenges of vehicular networks and Cloud*. New York, NY: Springer Science+Business Media, 2018.

[2] M. Chahal, S. Harit, K. K. Mishra, A. K. Sangaiah, and Z. Zheng, "A Survey on software-defined networking in vehicular ad hoc networks: Challenges, applications and use cases," *Sustainable Cities and Society*, vol. 35, pp. 830–840, Nov. 2017.

[3] J. C. Nobre *et al.*, "Vehicular Software-Defined Networking and fog computing: Integration and design principles," *Ad Hoc Networks*, vol. 82, pp. 172–181, Jan. 2019.

[4] I. Ku, Y. Lu, M. Gerla, R. L. Gomes, F. Ongaro, and E. Cerqueira, "Towards software-defined VANET: Architecture and services," in *2014 13th Annual Mediterranean Ad Hoc Networking Workshop (MED-HOC-NET)*, Slovenia, pp. 103–110, June. 2014.

[5] K. Shaaban, M. A. Khan, R. Hamila, and M. Ghanim, "A Strategy for Emergency Vehicle Preemption and Route Selection," *Arabian Journal for Science and Engineering*, vol. 44, no. 10, pp. 8905–8913, Oct. 2019.

[6] H. A. Ameen, A. K. Mahamad, S. Saon, D. Md. Nor, and K. Ghazi, "A review on vehicle to vehicle communication system applications," *IJEECS*, vol. 18, no. 1, p. 188, Apr. 2020, doi: 10.11591/ijeecs.v18.i1.pp188-198.

[7] A. M. de Souza, A. Boukerche, G. Maia, R. I. Meneguette, A. A. F. Loureiro, and L. A. Villas, "Decreasing greenhouse emissions through an intelligent traffic information system based on inter-vehicle communication," in *Proceedings of the 12th ACM international symposium on Mobility management and wireless access* Montreal, QC, Canada, pp. 91–98, 2014.

[8] W. Zhu, D. Gao, W. Zhao, H. Zhang, and H.-P. Chiang, "SDN-enabled hybrid emergency message transmission architecture in internet-of-vehicles," *Enterprise Information Systems*, vol. 12, no. 4, pp. 471–491, Apr. 2018.

[9] S. C. Litescu, V. Viswanathan, H. Aydt, and A. Knoll, "Information Dynamics in Transportation Systems with Traffic Lights Control," *Procedia Computer Science*, vol. 80, pp. 2019–2029, 2016.

[10] De Schutter, Bart, and Bart De Moor. "Optimal traffic light control for a single intersection." *European Journal of Control* 4, no. 3, pp. 260-276, 1998.

[11] D. Krajzewicz *et al.*, "Simulation of modern Traffic Lights Control Systems using the open source Traffic Simulation SUMO," *Proceedings of the 3rd industrial simulation conference*, p. 4, 2005.

[12] A. M. Turky, M. S. Ahmad, and M. Z. M. Yusoff, "The Use of Genetic Algorithm for Traffic Light and Pedestrian Crossing

Control," *International Journal of Computer Science and Network Security*, pp. 88–96, 2009.

[13] J. García-Nieto, E. Alba, and A. Carolina Olivera, "Swarm intelligence for traffic light scheduling: Application to real urban areas," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 2, pp. 274–283, Mar. 2012.

[14] M. Bani Younes and A. Boukerche, "An Intelligent Traffic Light scheduling algorithm through VANETs," in *39th Annual IEEE Conference on Local Computer Networks Workshops*, Edmonton, AB, Canada, pp. 637–642, 2014.

[15] Shridevi Jeevan Kamble, Manjunath R Kounte, "A Survey on Emergency Vehicle Preemption Methods Based on Routing and Scheduling", International Journal of Computer Networks and Applications (IJCNA), 9(1), PP: 60-71, 2022, DOI: 10.22247/ijcna/2022/211623.

[16] N. M. Z. Hashim, "Traffic Light Control System for Emergency Vehicles Using Radio Frequency," *IOSRJEN*, vol. 3, no. 7, pp. 43–52, Jul. 2013.

[17] K. A. Annan, E. O. K. Addo, J. A. Baidoo, and K. O. Anim, "Design and Implementation of an Emergency Traffic Light Pre-emption System Using PLC Automation," ijird, vol. 10, no. 1, Jan. 2021, doi: 10.24940/ijird/2021/v10/i1/157073-386715-1-SM.

[18] D. P. Sangeetha, S. Vijayashaarathi, Amandeep Kaur, N. Bavya, M. Janani, "Dynamic Traffic Light Switching based on Traffic Density", Annals of RSCB, pp. 1964–1972, May 2021

[19] A. A. Salih, A. Lewis, and E. Chung, "Dynamic Preemption Algorithm to Assign Priority for Emergency Vehicle in Crossing Signalised Intersection," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 518, p. 022038, Jun. 2019, doi: 10.1088/1757-899X/518/2/022038.

[20] S. Humagain, R. Sinha, E. Lai, and P. Ranjitkar, "A systematic review of route optimisation and pre-emption methods for emergency vehicles," *Transport Reviews*, vol. 40, no. 1, pp. 35–53, Jan. 2020, doi: 10.1080/01441647.2019.1649319.

[21] H. Mu, L. Liu, Y. Song, and N. Wang, "Control Strategy of Signal Transition after Emergency Vehicle Signal Preemption," *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1–11, Jan. 2020, doi: 10.1155/2020/1382415.

[22] P. N. Bideh, N. Paladi, and M. Hell, "Software Defined Networking for Emergency Traffic Management in Smart Cities," *IWVSC'2019*. Springer, p. 12, 2019.

[23] Zhong, Li, and Yixiang Chen. "A Novel Real-Time Traffic Signal Control Strategy for Emergency Vehicles." *IEEE Access* 10 (2022): 19481-19492.

[24] A. Rego, L. Garcia, S. Sendra, and J. Lloret, "Software Defined Network-based control system for an efficient traffic management for emergency situations in smart cities," *Future Generation Computer Systems*, vol. 88, pp. 243–253, Nov. 2018.

[25] M. B. Younes and A. Boukerche, "An efficient dynamic traffic light scheduling algorithm considering emergency vehicles for intelligent transportation systems," *Wireless Networks*, vol. 24, no. 7, pp. 2451–2463, Oct. 2018.

[26] Nazila Bagheri, Saleh Yousefi, and Gianluigi Ferrari. "Software-defined Control of Emergency Vehicles in Smart Cities. "

In *2020 10th International Conference on Computer and Knowledge Engineering (ICCKE)*, pp. 519-524. IEEE, 2020.

[27] Rosayyan P, Paul J, Subramaniam S, Ganesan SI. “An optimal control strategy for emergency vehicle priority system in smart cities using edge computing and IOT sensors. ” *Measurement: Sensors*. 2023 Apr 1;26:100697.

[28] Rammohan, A. “Revolutionizing Intelligent Transportation Systems with Cellular Vehicle-to-Everything (C-V2X) Technology: Current Trends, Use Cases, Emerging Technologies, Standardization Bodies, Industry Analytics and Future Directions. ” *Vehicular Communications* (2023): 100638.

[29] Dijkstra's Shortest Path Algorithm | Brilliant Math & Science Wiki. [online] Available at: https://brilliant.org/wiki/dijkstras-short-path-finder [Last Accessed 13 Feb. 2020].

[30] OMNET++ simulator. [online] Available at: https://omnetpp.org [Last Accessed 13 Feb. 2020].

[31] SUMO simulator. [online] Available at:http://sumo.sourceforge.net [Last Accessed 13 Feb. 2020].

[32] VEINS simulator. [online] Available at: https://veins.car2x.org [Last Accessed 13 Feb. 2020].

[33] OSM. [online] Available at: https://www.openstreetmap.org/#map=16/38.0775/46.2885 [Last Accessed 15 Oct. 2018].