


ORIGINAL PAPER

Open Access



Multi-agent preemptive longest queue first system to manage the crossing of emergency vehicles at interrupted intersections

Ali Louati^{1,2*} , Sabeur Elkosantini², Saber Darmoul³ and Hassen Louati⁴

Abstract

Favouring the crossing of Emergency Vehicles (EVs) through intersections in urban cities is very critical for people lives. There have been several efforts toward developing Traffic Signal Control Systems (TSCS) dedicated to control efficiently the traffic flow, but few are the efforts toward developing Traffic Signal Priority Systems (TSPS) dedicated to favour the crossing of EVs (such as ambulances, firefighters, police cars, etc.). Multi-Agent Systems were considered to develop several distributed TSCS, while very few works have developed distributed TSPS. Such systems lack on dealing with the EVs crossing issues while maintaining a fluid state of the traffic. In the literature, the Longest Queue First – Maximal Weight Matching (LQF-MWM) is proved to guarantee a stable TSCS. Recently, the LQF-MWM technique is increasingly used to benchmark and assess adaptive TSCS. Moreover, the preemption is one of the most effective techniques used to prioritise the crossing of EVs. This paper is the first to rely on LQF-MWM assumptions, preemption technique, and Multi-Agent Systems to design a distributed TSPS. The suggested system has two main purposes, which are the guidance of EVs and the control of traffic signals. Nine agents are implemented to govern a network of nine intersections, where each agent uses the Multi Agent System based Preemptive Longest Queue First – Maximal Weight Matching. We have referred to VISSIM traffic simulation software for benchmarking and analysis. To assess the suggested system, we have developed a distributed and preemptive version of VISSIM Optimized Stage-Based Fixed-Time algorithm. Python is considered to develop the suggested systems, and Spade platform is considered as agents' platform. Several Key Performance Indicators are considered to assess the performance of all controllers including delay time, travel time, vehicles queue occupancy, number of stops, distance traversed, and speed. Experimental results show a competitive performance of the developed system to maintain a fluid traffic and guide efficiency EVs.

Keywords: Traffic signal control systems, Traffic signal priority systems, Emergency vehicle, Multi-agent systems, Preemption technique, LQF-MWM, Optimized stage-based fixed-time controller

* Correspondence: a.louati@psau.edu.sa; ali.louati@hotmail.com

¹Information System Department, Prince Sattam bin Abdulaziz University, 11942 Alkharj, Kingdom of Saudi Arabia

²University of Tunis, ISG, SMART Lab, 41, Avenue de la Liberté, Bouchoucha, Le Bardo, 2000 Tunis, Tunisia

Full list of author information is available at the end of the article

1 Highlights

- A distributed, fully heterarchical and adaptive Traffic Signal Priority System is designed;
- A new preemptive algorithm based on a state-of-the-art algorithm named the Longest Queue First – Maximal Weight Matching (LQF-MWM) algorithm;
- Performance is assessed against a customized distributed and preemptive adaptation of the VISSIM Optimized Stage-Based Fixed-Time algorithm (OSBFX);
- Competitive performance in case of stable and unstable traffic volumes.

2 Introduction

In an inherently non-static environment, vehicular traffic control on road networks became a complex decision making task. Two hundred ninety-five million vehicles per hour are delayed due to traffic signals [9]. Traffic in urban areas has many problems such as congestions, bottlenecks due to peak hours, accidents, clearing the way to high priority vehicles. According to the Highway Traffic Safety Administration & Department of Transportation [11], 35,092 people died due to accidents in the US in 2015, while thousands of people are dying every day due to emergency vehicles (EVs) delays.

2.1 An overview on traffic signals control and priority systems

Several Traffic Signal Control Systems (TSCS) were developed in the literature to maintain traffic fluidity at signalized intersections. Such systems belong to two broad classes which are; the traditional control strategy, named fixed-time or pretimed control such as TRANSYT [29] and MAXBAND [16], and adaptive traffic control strategy such as SCOOT [30], CRONOS [3], and RHODES [23]. In spite of the high number of the developed TSCS, very few works have given attention to guide EVs to cross smoothly and efficiency an isolated intersection [4, 14] or multi signalized intersection(s) [12, 28, 34, 41]. Previous researchs have focused on the developing of adaptive and pretimed control techniques without considering signal priority treatments [22]. Insuring safety especially when disturbances occur, become a major concern of efficient and effective TSCS which is responsible of determining decisions to be sent to traffic signals. These decisions should improve traffic flow and help in protecting people lives.

Traffic Signal Priority Systems (TSPS) are giving less attention in the literature compared to TSCS. Such systems can be classified into six categories [40] which are; pre-emption versus priority (pre-emption

can be considered as the highest level of priority), one approach versus conflicting approaches, active transit signal priority versus passive transit signal priority, signal mode versus multiple mode, signal intersection versus coordinated intersection, and one request at a time versus multiple request. The majority of TSPS in the literature relies on electronics technology allowing the detection of EVs, the communication between those vehicles and traffic signals. Such technology includes Wireless Sensor Network (WSN) [4], RFID [31], national transportation communications for Intelligent Transport System protocol (NTCIP) [5], centralized server that supportes real traffic information and calculates the shortest path for EVs to pass by [13]. Shibuya, et al. [32] have presented an attempt for traffic management that supports EVs, called FAST emergency preemption system. Authors here have focused on the technological aspect to deal with the problem by introducing the infrared beacon as a key of a traffic signal control structure. Mirchandani & Lucas [24] introduced a transit signal priority and rail emergency preemption. The system suggested by authors, referred as Categorized Arrivals based Phase Reoptimization at Intersections (CAPRI), integrates a dynamic programming based real time traffic signal control. Despite their success and popularity, these technologies still face some limitations, including the fact that they did not consider traffic loads and adaptation of traffic light per traffic load. Other works did not focus upon prioritization between EVs. Accordingly, decision needs refinement to make some changes on the decision process.

2.2 An overview on intelligent control approaches

With regard to artificial intelligence techniques, few works have given attention to EVs issues when developing TSCS. The majority of these works used the preemption technique, such as, Huang, et al. [14] that integrated preemption rules into a Petri Nets based system to allow queued vehicles with the same direction of the EV to cross the intersection. Furthermore, Qin & Khan [28] adopted two-phases algorithm that enables signal transitioning from normal operation to EV signal preemption and vice versa. The first phase is a relaxation method, while the second phase is a step-wise search strategy. The preemption mechanism switches from a current phase to a new selected phase once an EV is detected by sensors, through a preemption device located in the vehicle. This device has as role to send a request to the preemption system asking to favour the crossing of the EV in which is located. Houli et al. [12] used Reinforcement Learning technique to develop a multiobjective control algorithm for urban traffic signal control. The proposed system

needs intelligent vehicles equipped with vehicular ad hoc network communication devices. The exchange of information regarding traffic and vehicles is based on a vehicular ad hoc network. The objectives investigated in [12] include vehicle stops, average waiting times, and the maximum queue length within intersections. Marcianò et al. [20] developed a signal setting model based genetic algorithm that takes into consideration the vehicular flow and the congestion at intersections. In this latest work, authors have dealt with an evacuation case. They have developed a dynamic model based on a path choice model by using a behavioural rules of different users. However, they have just considered urban evacuation as emergency without considering EVs types.

With regard to the EV types, very few works have specified the type of the EV, such as buses [10, 12, 15], police cars [32], ambulances [4, 14]. While other works have merged several types of EV at the same time, such as in [39] where the authors have defined an EV as a vehicle that need priority to serve public needs, like fire trucks, ambulance, police cars, or in [24] where authors have considered buses, tram, police cars, and ambulance as EVs.

Recently, several research efforts focused on developing distributed TSCS using multi-agent technology [7], but not on developing distributed TSPS. In this paper, the main reasons for considering multi-agent systems (MAS) are as follows:

- This work aims to enable complex problem modeling and resolution in order to achieve a distributed control;
- The MAS paradigm alone is not able to achieve capabilities specifically dedicated to EVs management and the reason behind that is because of their generic conceptual framework and lack of built-in adaptation mechanisms [7];
- Several computational intelligence techniques were combined with MAS to achieve reactive and adaptive control of traffic signals [8, 33];
- Despite their success and popularity, these approaches still face some limitations, including control of disturbances related to EVs and assessment of performance under real life situations [21].

2.3 Problem statement

This article relies on the Longest Queue First – Maximal Weight Matching (LQF-MWM) [37, 38] as a design guideline. We are referring to the LQF-MWM to design an intelligent TSPS capable to control traffic signals and favour the crossing of EV for several reasons, which are:

- The LQF-MWM is adopted often to control traffic signals [37, 38];
- It helps in creating and improving data base of signal control decisions [18];
- It provides easy adaptation to cover and control signals at a single intersection [17, 19], and at a network of multi-intersections [7]. Fourth, it is considered as adaptive and capable to stabilise vehicles queues [1];
- According to Wu et al. [36], LQF-MWM showed a competitive performance compared to several approaches existing in the literature, such as the First Come First Serve policy (FCFS) and Colony System (ACS);
- Several recent references have used the LQF-MWM for comparison purposes and assessment [1, 6, 36];
- The experiment in [7] shows that the LQF-MWM is able to deal with a disturbed network made of multiple intersections;
- The LQF-MWM promotes the minimization of both queue lengths and vehicles delay [1, 18].

Several limitations rises from the cited references:

- There is no reference that have considered prioritisation between vehicles and all vehicles are treated uniformly;
- It is assumed that the preemption is the most used technique to prioritise the crossing of EVs, but this technique is not yet integrated into the LQF-MWM;
- The majority of works that have referred to LQF-MWM have considered a single intersection, while those whom have considered more complex network of several intersections lack in communication technique and cooperative control between signals, in order to ensure a fluid network;
- There is no reference that have integrated the preemption technique into MAS;
- They lack in-depth analysis and assessment;
- There is a need for systems that are able to maintaining the traffic fluidity while favouring the crossing of EVs.

To overcome these limitations, developing an intelligent TSPS that ensure safety and improve traffic fluidity is a challenging problem for which adaptation mechanisms need to be developed to deal with EVs in an intelligent way. This article addresses the MAS, the preemption technique, and the LQF-MWM approach to develop an adaptive TSPS to control traffic signals and favour the crossing of EV at signalized intersections. An effective model of communication and collaboration between agents that guarantees an efficient

decision making is introduced. A novel hybridization and a detailed comparative analysis using innovative and technological implementation choices are provided.

The remainder of this article is organized as follow: Section 3 details the modelling and the concepts of the suggested system. Section 3 introduces the adaptation and the control algorithms toward developing the suggested system. Section 5 details the architecture of the system. Section 6 details the tools and the technologies adopted for the development of all systems considered in this work. Section 7 describes the experimental framework. Section 8 analysis and assess the performance of all controllers. Finally, Sections 9 concludes the paper.

3 Preemptive LQF-MWM control algorithm

The preemptive LQF-MWM (P-LQF-MWM) should maintain the fluidity of the traffic at acceptable level if there is EVs or not. Congestion, bottlenecks, unbalanced traffic load, high traffic load, and more can disturb EVs to cross smoothly intersections. Indeed, the suggested system generates suitable decisions in order to eliminate non-desired traffic status and solve emergency cases. The algorithms are detailed in the following subsections.

3.1 The basic LQF-MWM algorithm

To develop the P-LQF-MWM the LQF-MWM algorithm proposed by Wunderlich et al. [37, 38] is considered. The LQF-MWM algorithm gives priority to approaches with the longest vehicle queues. It is based on the following assumptions:

- At each approach of the intersection, the amount of traffic does not exceed the capacity of the approach;
- It is not allowed to overload any of the destination approaches;
- All queues are served in accordance with the policy fixed by the signal control algorithm;
- At time $t = 0$, there is no loopback traffic, all $Q(t_0) = 0$;
- The evolution of the queue occupancy can be expressed as:

$$Q(t + 1) = Q(t) + A(t) - D(t) \tag{1}$$

- The weight produced by the LQF-MWM algorithm at time t is given by:

$$W(t) = \langle Q(t), S_{j,k}(t) \rangle = \sum_{j,k} Q_{j,k}(t) S_{j,k}(t) \tag{2}$$

- The average rate of vehicles moving through the intersection from input approach j destined for output approach k .

The LQF-MWM algorithm is summarized in Fig. 1.

3.2 Preemptive LQF-MWM algorithm

The suggested algorithm referred as P-LQF-MWM includes the following steps:

```

Algorithm: LQF-MWM


---


Input: Queue occupancies  $Q(t)$ 
Output: Decision (Phase sequence)
Begin
  For each  $T$  time
    For each approach
      If input approach  $j$  is selected by the matching matrix to be connected to
        output approach  $k$ 
        Measure the queue occupancy  $Q(t)$  of vehicles currently queued at
        the approach  $j$ 
        Calculate the sum  $W(t)$  of produced weights according to (2)
        Phase to be served  $\leftarrow$  the approach with the highest weight
        Return Phase to be served
      Endif
    End For
  End For
End
    
```

Fig. 1 LQF-MWM algorithm

- **Step 1** - Reception of the local traffic data at each time t ;
- **Step 2** – Use the LQF-MWM (see Section 3.1) to control traffic under normal situation;
- **Step 3** – Analyze data in order to detect possible EV(s);
- **Step 4** – If EV(s) is/are located at the local intersection:
 - **Step 4.1** – Analyze the EV(s) information (number of EVs, type(s), destination(s));
 - **Step 4.2** – According to the number of EVs, create a local decision (phase sequence) using the following cases:
 - **Step 4.2.1** – If there is more than one EV:
 - **Step 4.2.1.1** - If the EVs have different types: serve the EV with the highest priority, then the lowest, and finally the normal priority;
 - **Step 4.2.1.2** - If the EVs have the same type:
 - Step 4.2.1.2.1** – Analyze the local queue occupancies;
 - Step 4.2.1.2.2** - Serve the EV located at the approach having the lowest queue occupancy and the nearest to the signal head;
 - **Step 4.2.2** – If there is just one EV: Serve it;
 - **Step 4.3** – Request the traffic fluidity data of the neighbouring intersection(s);
 - **Step 4.4** – Reception of the traffic fluidity data from the neighbouring intersection(s);
 - **Step 4.5** – Evaluate and compare the traffic fluidities of the neighbouring intersection(s) (queue occupancies);
 - **Step 4.6** – Create a global decision: choose the next intersection that will receive the EV according to two factors; the EV destination and the collected traffic fluidities data of neighbours. The

intersection with the lowest queue occupancies will be chosen;

- **Step 4.7** – Outputs: deliver the control decision (local and global decisions) to be applied by the agent;

- **Step 5** – Go to step 1.

The inputs and outputs used by the suggested algorithm which are described in the previous steps are highlighted in Fig. 2.

4 System modelling and concepts

In this section, a detailed description is provided of the system modelling and concepts including emergency cases, control decision, EVs detection, and priority rules related to EVs types.

4.1 Assumptions

In this work, three types of EV priority are denoted, which are:

- Type 1 'HS': this type of EV is considered as the highest priority. For example ambulance;
- Type 2 'H': this type of EV is considered as the high priority. For example fire-trunk;
- Type 3 'N': this type of EV is considered as the normal priority. For example Police car.

The MAS-P-LQF-MWM has two main tasks:

1. Monitor the traffic and adapt both phase durations and sequences to variations of traffic volumes;
2. Prioritize the crossing of EVs whenever located at any of the approaches.

Let's define the following nomenclature:

- I is the number of intersections (9 intersections);
- IN_i ($i = 1, \dots, I$) is the ID of an intersection;
- M is the number of approaches at an intersection (4 approaches per intersection);

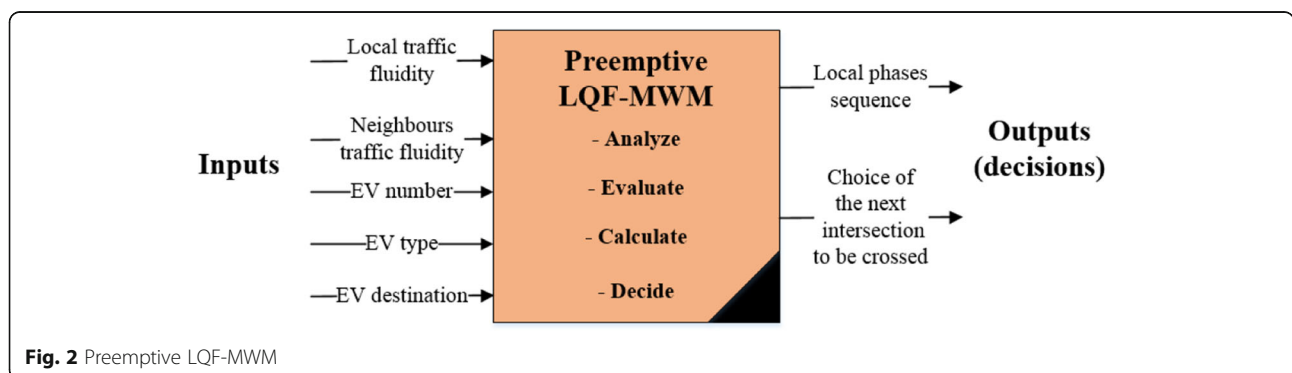


Fig. 2 Preemptive LQF-MWM

- $AP_m (m = 1, \dots, M)$ is the ID of the approach m ;
- T_{EV} is the type of the EV;
- P is the priority rule assigned to the EV (HS = highest priority; H = high priority; N = normal priority);
- $DI_i (i = 1, \dots, I)$ is the destination intersection i that an EV should reach;
- $DAP_m (m = 1, \dots, M)$ is the destination approach m that an EV should reach;
- DS_t is the distance to the signal head of an EV at the detection instant t ;
- OR_m is the order of the phase associated to approach m ;
- NI is the next chosen intersection to receive the EV;
- Q_m is the queue occupancy of an approach m ;
- $Q(t) = Q_m(t) (m = 1, \dots, M)$ is the queue occupancy vector represented in number of vehicles currently queued at time t ;
- $S_{jk}(t)$ is the allowable intersection configurations considered by the algorithm where; $S_{jk}(t) = 1$ if input approach j is selected by the control algorithm to connect to output approach k ; otherwise $S_{jk}(t) = 0$;
- $W(t)$ is the weight produced by the LQF-MWM algorithm at time t ;
- $D(t)$ is the number of vehicles departed from approach i to approach j during time slot t ;
- $A(t)$ is the number of vehicles arriving to the queue at time t ;

4.2 Emergency case representation

An emergency case is a vector of attributes (see Table 1) related to one or more EV(s) located at any approach of a signalized intersection. These attributes include indicators related to the ID of the approach where the EV is detected, the type of the EV, its destinations (intersection and approach), the distance to the signal head at the moment of detection, and the queue occupancy of its approach.

Table 2 illustrates a detailed example of an emergency case of an intersection which has EVs detected at its approaches.

According to Table 2:

- An ambulance with highest priority is located at the approach 4 of intersection 1. The ambulance should reach a hospital at the approach 3 of intersection 7 (see Fig. 3). This vehicle is 350 m from the signal head, and its approach has 15 vehicles queued.
- Approach 2 of intersection 1 has no EV;

Table 2 Example of an emergency case

IN_i	AP_m	T_{EV}	P	DI_i	DAP_m	DS_t	Q_m
1	4	Ambulance	HS	7	3	350	15
1	2	NULL	NULL	NULL	NULL	NULL	NULL
1	1	Fire – trunk	H	5	NULL	NULL	NULL
1	3	Ambulance	HS	6	2	220	10

- A fire-trunk with high priority located at the approach 1 of intersection 1. The vehicle should reach an accident located at the middle intersection which is number 5 (see Fig. 3).
- Another ambulance with highest priority is located at the approach 3 of intersection 1, the vehicle should reach a fire in a forest located at the approach 2 of intersection 6 (see Fig. 3). This vehicle is 220 m from the signal head, and its approach has 10 vehicles queued.

4.3 Control decision representation

The control decision is a vector of two attributes that should solve a detected emergency case (see Section 4.2). Table 3 illustrates the representation of the control decision, which includes a local and a global decisions. The local decision is related to the phase sequence of an intersection signals, while the global decision represents the next intersection that should receive the EVs.

As an example, we consider the emergency case described in Section 4.2 and we assign a control decision to this emergency case (see Table 4).

According to Table 4, the values of the local decision means:

- Approach 4 of intersection 1 is the second to be served with a green light;
- Approach 2 of intersection 1 is the fourth to be served with a green light;
- Approach 1 of intersection 1 is the third to be served with a green light;
- Approach 3 of intersection 1 is the first to be served with a green light.

The values of the global decision means:

- The ambulance located at the approach 4 of intersection 1 should take intersection 2 as next intersection toward its destination;
- Nothing for approach 2 of intersection 1;
- The fire-trunk located at the approach 1 of intersection 1 should take intersection 4 as next intersection toward its destination;

Table 1 Emergency case representation

IN_i	AP_m	T_{EV}	P	DI_i	DAP_m	DS_t	Q_m
--------	--------	----------	-----	--------	---------	--------	-------

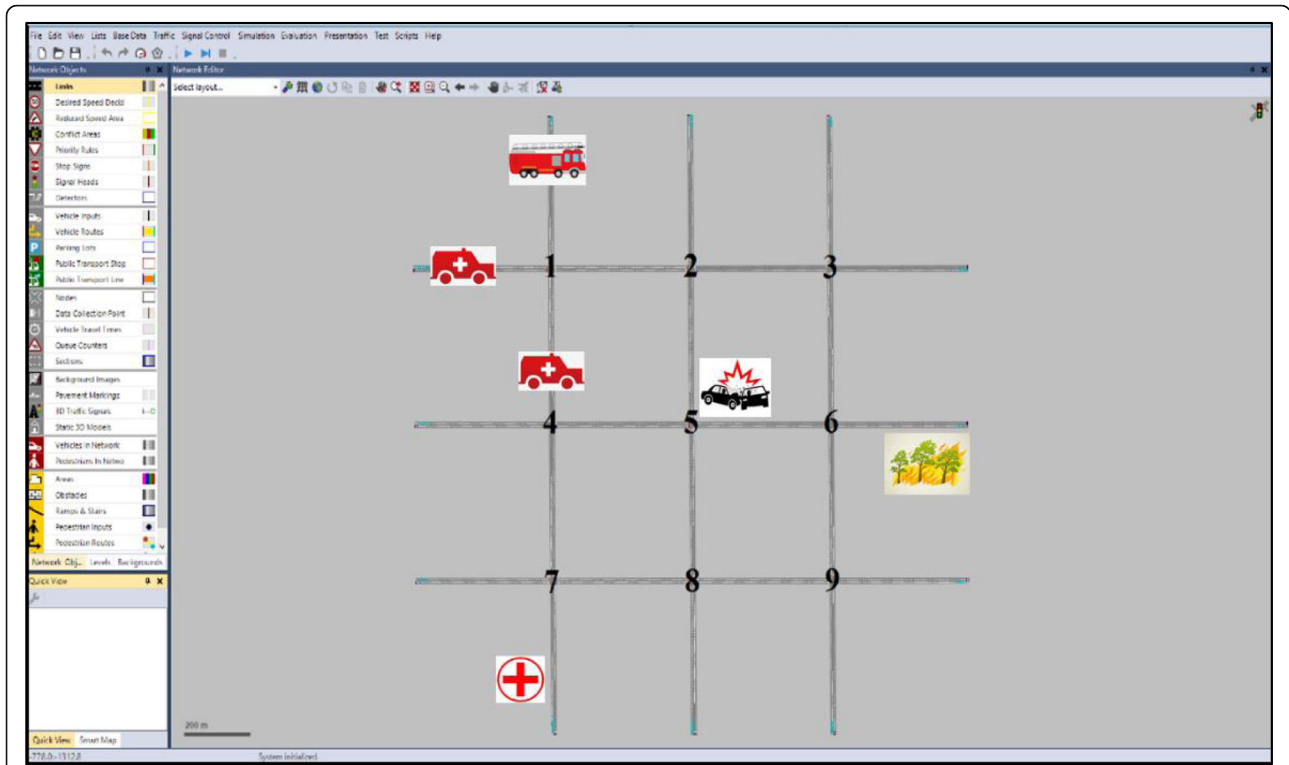


Fig. 3 The simulated network with illustrative scenarios

- The ambulance located at the approach 3 of intersection 1 should take intersection 2 as next intersection toward its destination.

4.4 Detection and creation of emergency case

The EVs are detected using sensors placed at the beginning of the North, South, East, and West approaches of the network (see Fig. 3). Once an EV penetrates any of these approaches, an emergency case (see Section 5) is created for all approaches of the intersection. The creation of the emergency case contains the following steps:

1. Get data from sensors placed at the beginning of each approach;
2. Analyze data;
3. In case of EV(s) detection at any approach of the intersection, create an emergency case according to the following cases:
 - a. Case 1: If one EV is detected at just one approach: fill the emergency case

with the intersection ID, the approach ID, the type of the EV, and its destination. The rest of the emergency case attributes are giving *NULL*;

- b. Case 2: If there is more than one EV having different types: fill the emergency case with the intersection ID, the approach ID, the type of each EV, their priorities, and their destinations. The rest of the emergency case attributes are giving *NULL*;
- c. Case 3: If there is more than one EV having the same type: fill the emergency case with the intersection ID, the approach id, the type of EVs, their priorities, their destinations, their distances from the signal head, and the queue occupancies of the EVs approaches.

A graphical illustration of the suggested concepts is presented in Fig. 4.

5 Heterarchical multi-agent system based P-LQF-MWM

The architecture provided in this paper is fully heterarchical, flat, and no hierarchy between agents. We associate an agent to each intersection, which communicates and coordinates with its adjacent

Table 3 Control decision representation

Emergency case										Control decision	
										Local decision	Global decision
IN_i	AP_m	T_{EV}	P	DI_i	DAP_m	DS_i	Q_m	OR_m		NI	

Table 4 Example of control decision

Emergency case								Control decision	
								Local decision	Global decision
IN_i	AP_m	T_{EV}	P	DI_i	DAP_m	DS_t	Q_m	OR_m	NI
1	4	Ambulance	HS	7	3	350	15	2	2
1	2	NULL	NULL	NULL	NULL	NULL	NULL	4	NULL
1	1	Fire – trunk	H	5	NULL	NULL	NULL	3	4
1	3	Ambulance	HS	6	2	220	10	1	2

neighbours by exchanging two main information. The first information is related to the traffic fluidity, while the second information is related to the existence of an EV at any of the intersection approaches. The decision making process of an agent is based on three main aspects; the P-LQF-MWM algorithm (Section 3.1), the data collected from the intersection, and the information exchanged with adjacent neighbours. The control architecture of the suggested MAS based preemptive LQF-MWM (MAS-P-LQF-MWM) is illustrated in Fig. 5.

5.1 Agent behaviours

In the suggested system, each agent relies on three behaviours, which are:

- P-LQF-MWM behaviour (see Section 3.2): this behaviour allows the agent to control its intersection and deliver control decisions using the data of the intersection and the data received from the Receiver behaviour;
- Sender behaviour: this behaviour takes as input the global decisions delivered by the P-LQF-MWM behaviour. Then it creates messages with destination addresses and send them to neighbouring agents;
- Receiver behaviour: this behaviour reads messages received from the neighbouring agents. Then, it extracts the order from the messages, and place the order at the P-LQF-MWM behaviour.

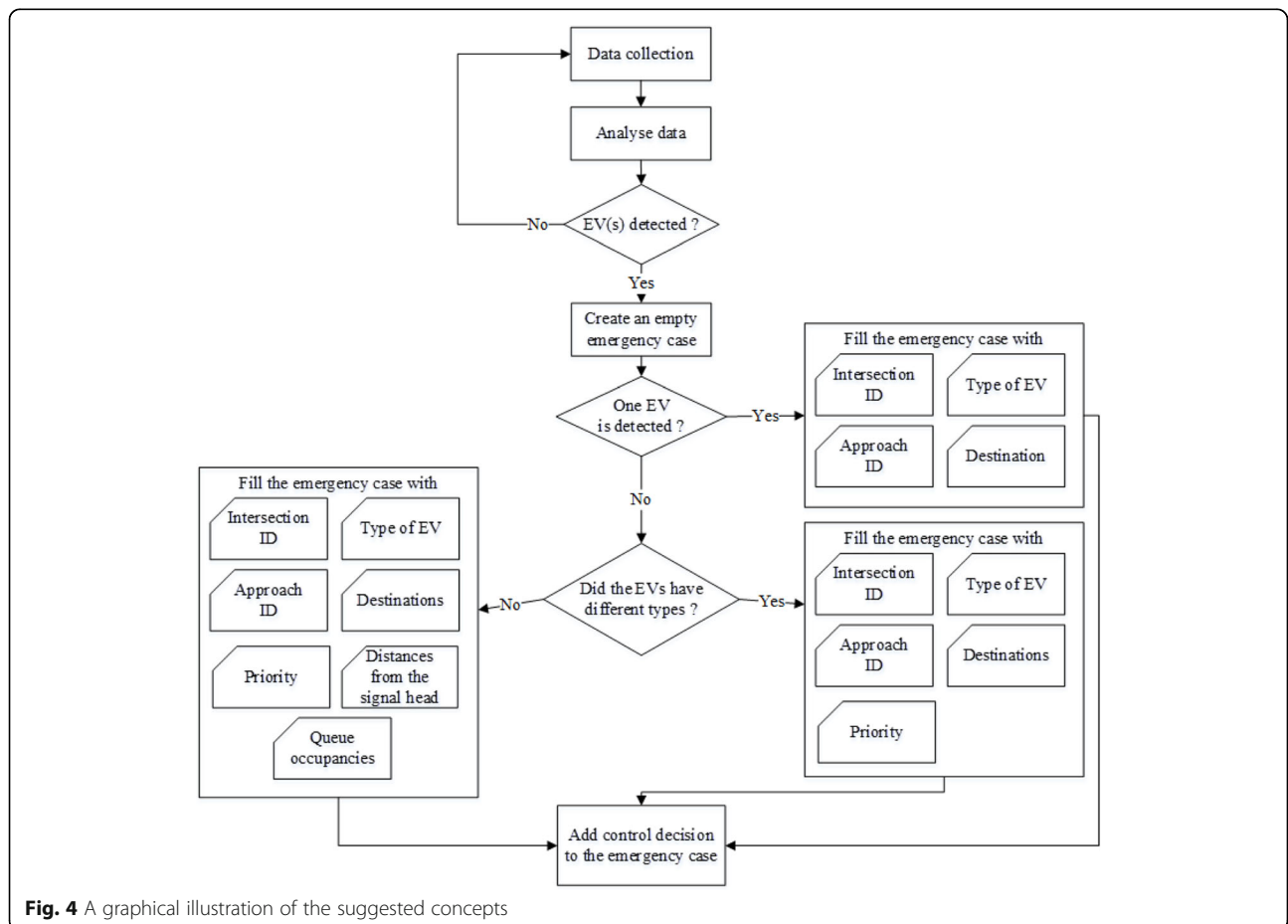


Fig. 4 A graphical illustration of the suggested concepts

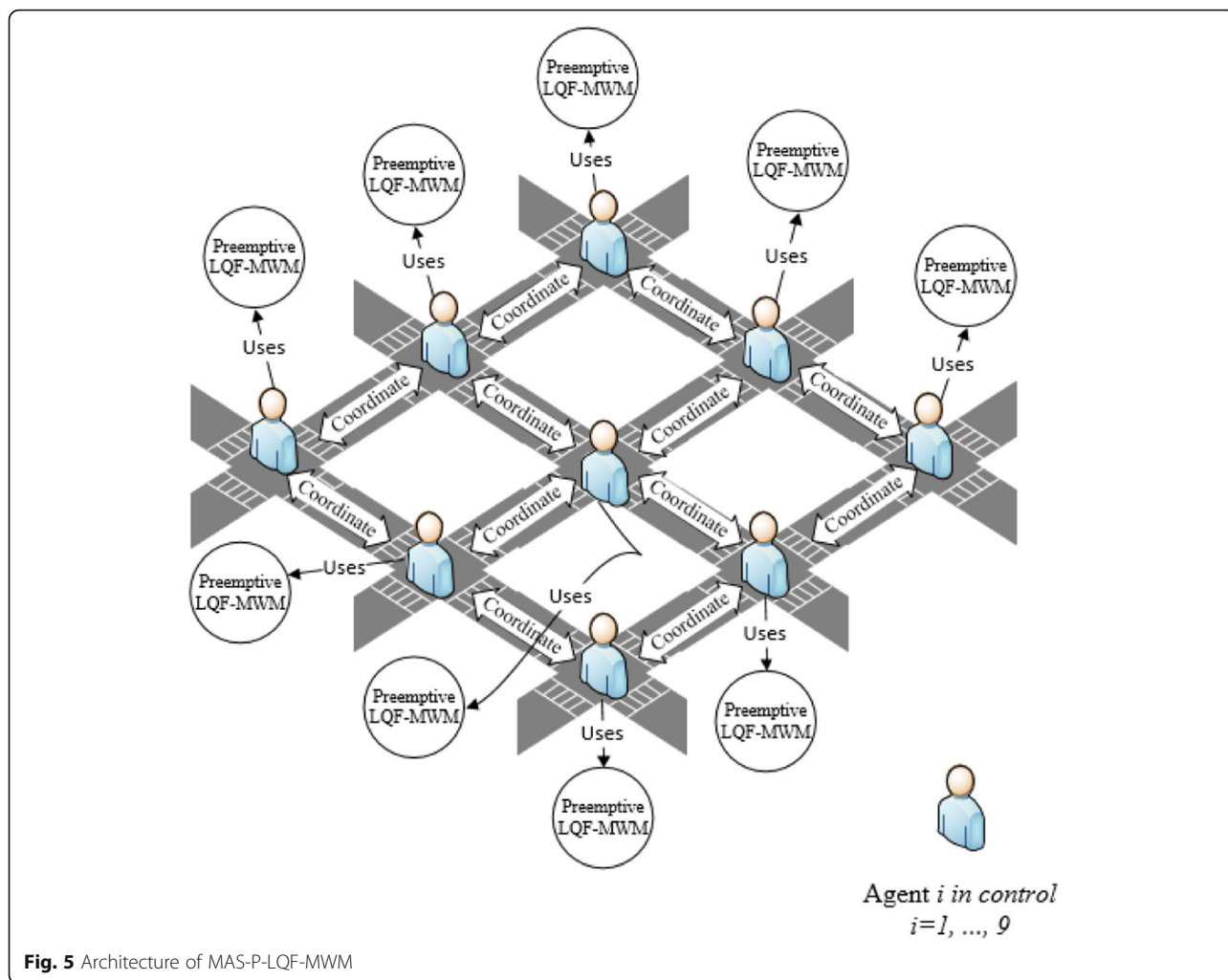


Fig. 5 Architecture of MAS-P-LQF-MWM

More technical details in Section 6.3 about the operation of agents behaviours, and more details about agent coordination in Section 5.2.

5.2 Agent coordination

In the suggested system, the neighbours of an agent x are those sharing with it a common approach. Agent x communicates with its neighbours whenever it detects at its intersection one or more EV(s). On an event of an EV located at its intersection, Agent x collects data from this vehicle (number of EV(s), type, and destination(s)). This data will be used as a part of the P-LQF-MWM inputs (see Section 3.2). Then, agent x communicates with its neighbours asking for their traffic fluidity information (queues occupancy). These information are collected by agent x and used to complete the P-LQF-MWM inputs. Whenever the P-LQF-MWM made a control decision, agent x applies it and informs the chosen neighbour that is about to receive the EV. Figure 6 details the agent state transition diagram.

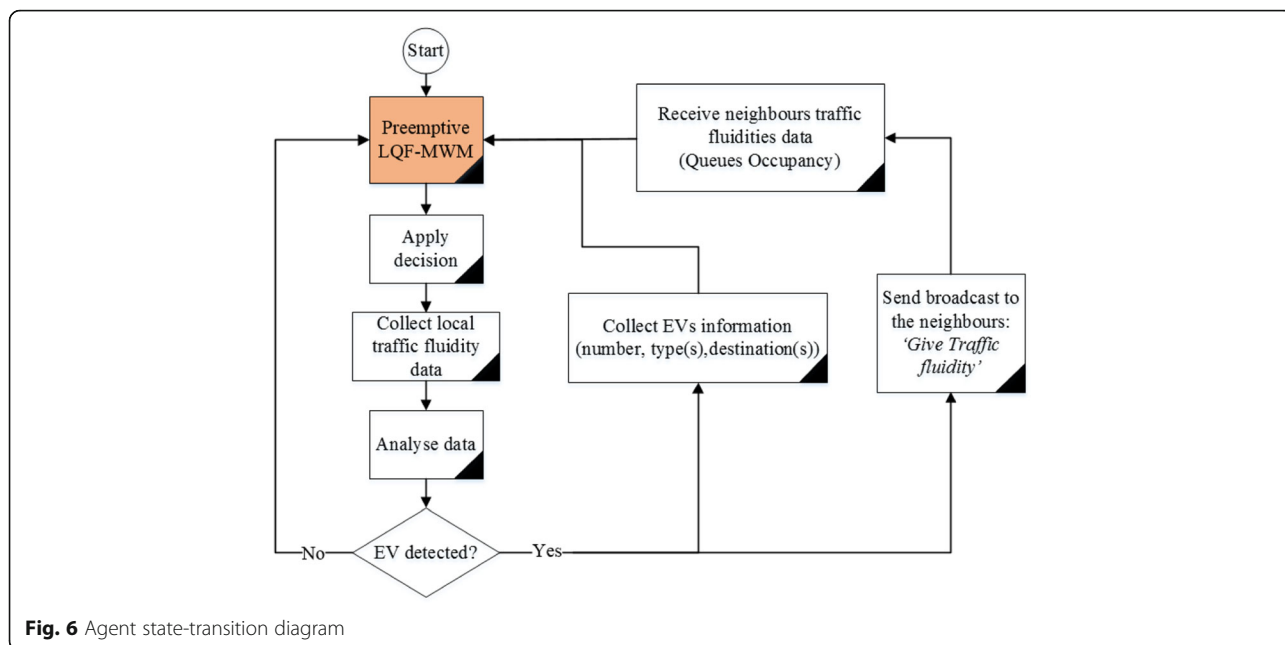
6 Tools and technologie adopted

In this section, we describe how agents are implemented, the tools used for implementation, and the software considered for the simulation and assessment.

6.1 VISSIM traffic simulator and sensors setup

VISSIM is a powerful state-of-the-art traffic simulation software. It is the leading simulation program for modelling multimodal transport operations and is being used worldwide by public sector, consulting, and universities [26]. It offers a module called Component Object Model (COM) that provides the preparation of data, the efficient control of examined scenarios, the inclusion of control algorithms, and the access to all network object attributes.

With regard to sensors technology, VISSIM offers several types of sensors that can be positioned according to the user need. We highlight three main sensors used in our simulations. The first type of sensors allows the detection of new EV entry and provides data about this vehicle. This type of sensors is placed at



the beginning of each approach. The second type of sensors collects data regarding queue occupancies, vehicles stops and delays at each approach. This type of sensors is placed at the signal head of each approach. The third type of sensors measure the travel time of EVs. This type of sensors has two sub-sensors each, one is placed at the beginning of each approach and the second one is placed at the end of all other approaches. However, in a real world implementation, inductive loops technology is the cheapest technology

[25] that can be used to determine vehicle queue occupancy, vehicle stops and delays, while RFID technology is the cheapest technology that can be used for the detection and tracking of EVs [31].

6.2 Agent implementation tools

To implement the suggested MAS-P-LQF-MWM, Python programming language is considered. Python provides a very powerful programming language for direct execution in the VISSIM context [26].

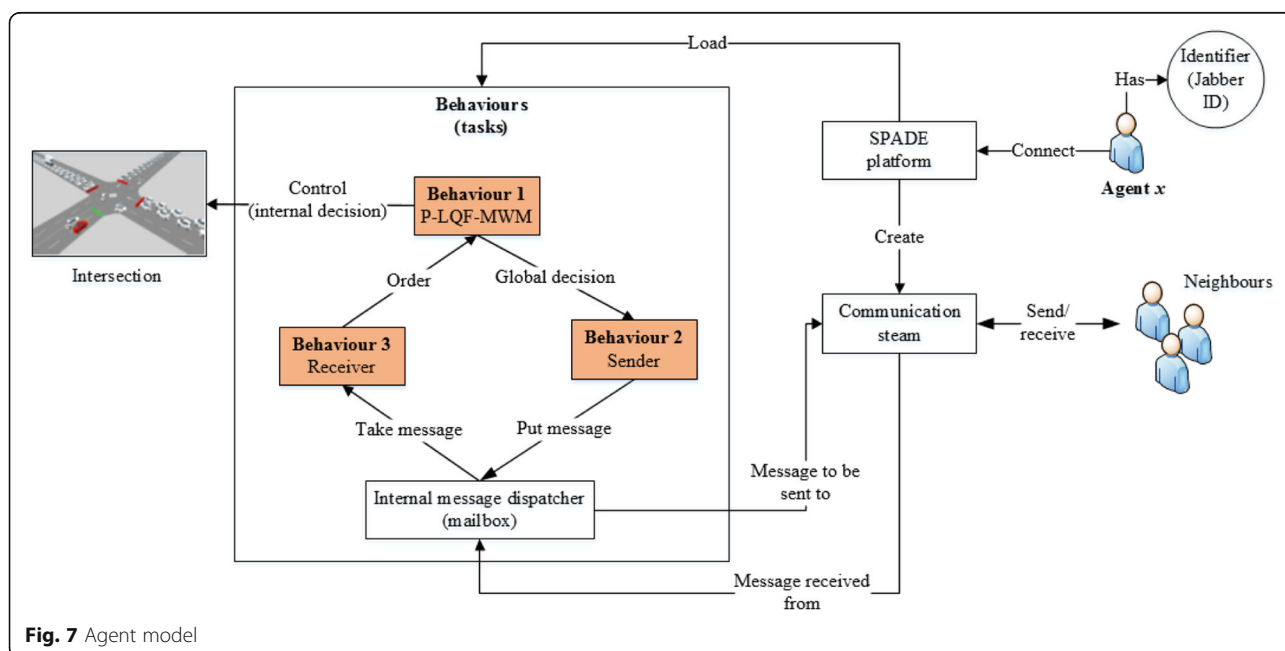


Table 5 Vehicles generated per hour at approaches for each scenario (AP_m refers to approach m)

Scenario	Intersection/ approach		Inter 1		Inter 2		Inter 3		Inter 4		Inter 5		Inter 6		Inter 7		Inter 8		Inter 9					
	AP1	AP2	AP3	AP4	AP1	AP2	AP3	AP4	AP1	AP2	AP3	AP4	AP1	AP2	AP3	AP4	AP1	AP2	AP3	AP4	AP1	AP2	AP3	AP4
S1	500	-	-	500	500	-	-	500	-	-	-	500	-	-	500	-	-	500	-	-	500	500	500	-
S2	400	-	-	600	350	-	-	800	750	-	-	650	-	-	900	-	-	850	-	-	900	-	400	500

To develop agents, Smart Python multi-Agent Development Environment called SPADE platform [27] is used. SPADE is based on XMPP/Jabber technology and it offers several facilities allowing the use of existing communication channel besides its extensible communication protocol based XML. Agents in SPADE are implemented as users and the platform where the agents communicate are implemented as servers.

6.3 Agent design

An agent has a unique identifier called Jabber ID and has three components (see Fig. 7), which are the connection mechanism to the platform, the message dispatcher, and a set of behaviours.

Agent x in Fig. 7 uses its Jabber ID to establish a connection with the platform. The Jabber ID includes the username, the address, and the server domain. After a successful registration, Agent x creates an open and persistent stream of communication with the platform, and loads its behaviours. As we can see in Fig. 7, Agent x has three behaviours, one for the suggested P-LQF-MWM to control its intersection (internal decision), one for sending messages and/or orders delivered by the P-LQF-MWM behaviour (queues occupancy and/or global decision), and the last one for reading and extracting messages and/or orders from other agents (queues occupancy and/or global decision). When a global decision is delivered by the P-LQF-MWM behaviour, the sender behaviour creates a message with a destination and puts it in the internal message dispatcher which works as a mailman. The dispatcher places the message in the Agent x communication stream in order to be sent to neighbouring agent(s). In case of order reception, the message sent to Agent x by a neighbouring agent is received by the communication stream. Then the dispatcher receives this message and places it in the receiver behaviour, which reads the message and extracts the order. Finally, the receiver behaviour places the order in the P-LQF-MWM behaviour in order to be used as inputs during the decision making process.

6.4 Code structure

In this section, a brief description is given of the classes considered to implement agents:

- **Agent.py:** in this file the agent class are implemented. This class contains the agent behaviours (see Fig. 7). The sender and receiver behaviours are explicitly implemented into the agent class, while the P-LQF-MWM behaviour recall *P-LQF-MWM.py* file;
- **P-LQF-MWM.py:** the suggested algorithm is implemented in this file;
- **Com.py:** this file establish the communication between VISSIM simulator and Python. Thus, it loads the intersections network, and it sets the preferences of the user regarding the simulation settings;
- **Decision.py:** this file allows to save the detected emergency cases (see Section 5) and the control decisions (see Section 4.3) giving by the suggested system during the simulation. This data is saved for future research purposes;
- **Main.py:** this is the file where the main program is implemented. This file works as runner allowing the initiation of VISSIM simulator (see Section 6.1), the connection with SPADE platform, and the starting of agents.

7 Experimentation

A network of nine signalized intersections are considered in this work (see Fig. 3). Each intersection has four approaches. We assigned two lanes and 400 m length for each approach [18, 37, 38]. In each approach vehicles can traverse the intersection according to the related green time and direction of the phases. Two lanes are associated to each approach, the first lane is for turning right and the second lane is a shared for straight and turning left movements.

7.1 MAS preemptive optimized stage-based fixed-time controller

A reference algorithm called the “Optimized Stage-Based Fixed-Time controller” (OSBFX) [26] is considered for the assessment of the suggested MAS-P-LQF-MWM. OSBFX algorithm is a VISSIM tool that allows the determination of the optimal phase duration of an intersection signals. First, the algorithm requires as input the desired cycle time and the number of phases of the intersection. In this manuscript, we have considered two different cycle time. The first cycle time is 60 s, and the

Table 6 Stops, speed, and distance results of EVs in scenario S1

EVs KPIs	MAS-P-LQF-MWM	MAS-P-OSBFX-60s	MAS-P-OSBFX-240 s
AVG No Stops	2.493	7.177	11.938
AVG speed	30.678	20.824	15.169
AVG distance traversed	1417.183	1784.533	5467.309

AVG: Average

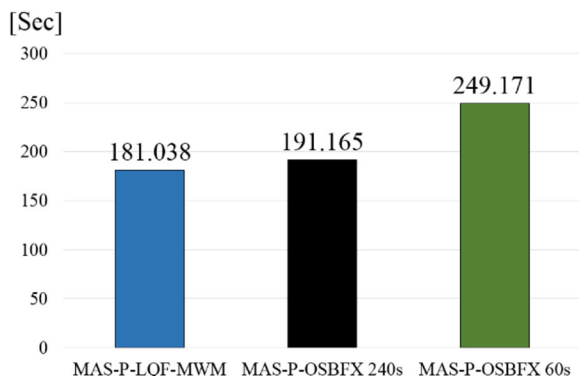


Fig. 8 Average total travel time per EV for scenario S1

second cycle time is 240 s, [2]. Second, VISSIM executes several simulations with different vehicles distribution in order to test different combinations of phase duration. Finally, VISSIM gives an optimized cycle time with an optimal phase duration for each approach of the intersection.

The requirements, methodology and assumptions of the OSBFX [26], are as following:

- VISSIM determines the average delay of all vehicles;
- For optimizing, the signal group in which the vehicles have the highest delay is determined for each stage;
- The stage with the lowest maximum average delay is selected as the best stage;
- The stage with the highest maximum average delay is selected as the worst stage;
- A second of green time is deducted from the best stage;
- A second of green time is added to the worst stage;
- If a second can no longer be deducted from the best stage, the second best stage is used. If this can no longer be shortened, the next worst stage is always taken iteratively. If no other stage can be shortened, the optimization is terminated;

- A signal program is considered to be better than another if one of the following criteria is met:
 - If the flow formed by the total number of vehicles driven through the node during the simulation run has increased significantly by at least 25 vehicles or by 10% if this is less;
 - If the flow has not significantly decreased by 25 vehicles or by 10% and the average delay across all vehicles has decreased;
- If a signal program is better than the best rated, it replaces this as the best. The optimization is continued with the next step;
- The optimization is terminated if one of the following criteria is met:
 - Once the signal program does not improve within 10 simulation runs;
 - Once the flow decreases by more than 25% compared to the best signal program;
 - Once the average delay increases by more than 25%.

For the particular needs of our assessment, we designed a distributed and Preemptive version of the OSBFX controller, referred as MAS-P-OSBFX. The preemptive module integrated within the OSBFX brakes the

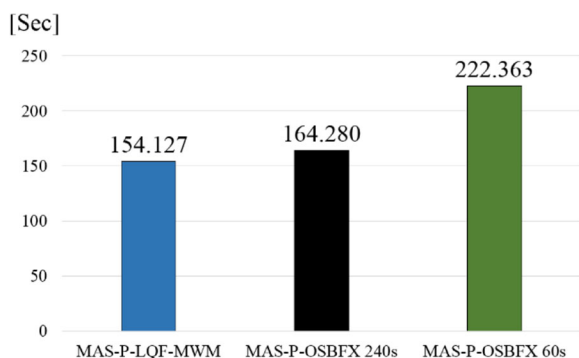


Fig. 9 AVG total delay per EV for scenario S1

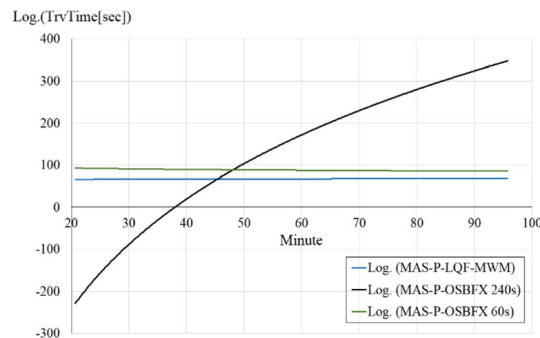


Fig. 10 Total travel time of all EVs for Scenario S1

current cycle time and assign green light to the approach where the EV is detected. The MAS-P-OSBFX is implemented in each agent to control its intersection and to favour the crossing of EVs. Each agent is autonomous and independent of other agents.

7.2 Simulated scenarios

Two scenarios are considered for the assessment with 3 h duration each:

- Scenario S1: has 12.000 generated vehicles. S1 allows the assessment of the performance of the controllers under stable traffic condition characterized by the same traffic density;
- Scenario S2: has 15.600 generated vehicles. S2 allows the assessment of the performance of the controllers under unstable traffic condition characterized by variable traffic loads.

Table 5 details the loads assigned to each approach per scenario.

7.3 Settings and KPIs

In this section, the different settings considered in the experiments are detailed:

- Regarding the probability distribution function used for vehicle arrivals, the default random seed value assigned by VISSIM which is 25 is used;
- Regarding the traffic demand, we used the dynamic assignment in VISSIM which gives a free choices to drivers to choose the path from their origin start until their final destination;
- Regarding driver behaviours, VISSIM uses the psycho-physical perception model of Wiedemann [35];
- Regarding the maximum speed limit, two different speed values are used which are:
 - 50 km/h as a desired speed distribution for ordinary vehicles which is a default value defined by VISSIM;
 - 90 km/h as a desired speed distribution for the EVs.

Moreover traffic data are collected form VISSIM as follows:

- Queue lengths are collected each 10 s;
- Vehicle stops are collected each 10 s;
- Delays are collected each 10 s;
- Travel times are collected each 10 s;

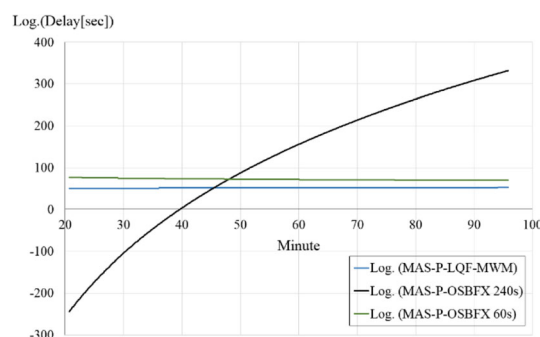


Fig. 11 Total delay of all EVs for Scenario S1

- Vehicles speed are collected each second.

The KPIs considered in this work are inspired from the works in [7, 17, 19]. To evaluate the controllers regarding EVs guidance, we investigated seven Key Performance Indicators (KPIs):

- The average number of stops of EVs. This refers to the number of times that EVs have stopped to reach their destinations. The algorithm who reduces the average number of EVs stops is considered the best;
- The average speed of EVs since located in the network until leaving it. The algorithm allowing the highest average speed for EVs is considered the best;
- The average distance traversed of EVs since located in the network until leaving it. The algorithm allowing EVs to reach their destinations with the minimum distance traversed is considered the best;
- The average total travel time per EV. This refers to the mean travel time spent by an EV since located in the network until reaching its destination. The algorithm who reduces the average total travel time of each EV is considered the best;
- The average total delay per EV. This refers to the time spent by an EV since located in the network until reaching its destination. The algorithm who reduces the average total delay of each EV is considered the best;
- The total travel time of all EVs. This refers to the total travel time of all EVs spent since located in the network until reaching their destinations. The algorithm who reduces the average total travel time of all EVs is considered the best;
- The total delay of all EVs. This refers to the total delay of all EVs spent since located in the network until reaching their destinations including queue delay. The algorithm who reduces the average total delay of all EVs is considered the best.

To evaluate the controllers regarding traffic fluidity, we investigated five KPIs:

Table 7 Stops, speed, and distance results of EVs in scenario S2

EVs KPIs	MAS-P-LQF-MWM	MAS-P-OSBFX-60s	MAS-P-OSBFX-240 s
AVG No Stops	1.114	31.716	5.305
AVG speed	39.851	4.855	11.319
AVG distance traversed	495.763	2885.373	1075.417
AVG Average			

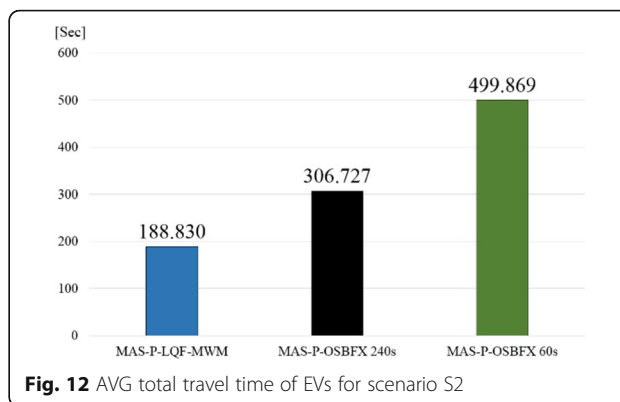


Fig. 12 AVG total travel time of EVs for scenario S2

- The average total travel time per vehicle in the network. This refers to the mean travel time of a single vehicle spent until leaving the network. The algorithm who reduces the average total travel time of all vehicles in the network is considered the best;
- The average total delay per vehicle in the network. This refers to the mean delay of a single vehicle spent until leaving the network. The algorithm who reduces the average total delay of all vehicles in the network is considered the best;
- The total travel time in the network. This represents the total travel time spent by all vehicles in the network. The algorithm who reduces the total travel time of all vehicles in the network is considered the best;
- The total delay in the network. This represents the total time spent by all vehicles in the network, the average queue delay of vehicles, and the time taken by vehicles to leave the network. The algorithm who reduces the total delay of all vehicles in the network is considered the best;
- The average queue occupancy per intersection. This represents the mean number of vehicles queued in each approach of the network. The algorithm who reduces the average queue occupancy in the network is considered the best;

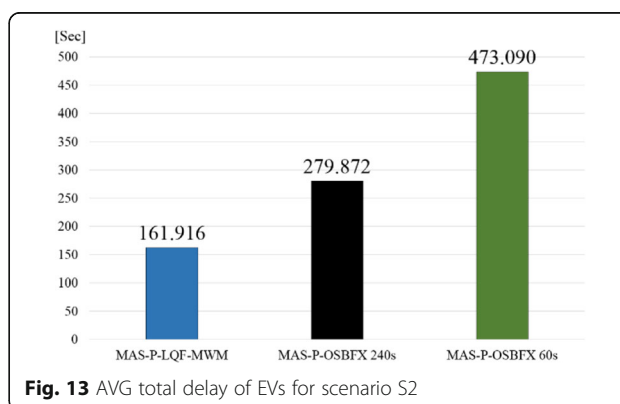
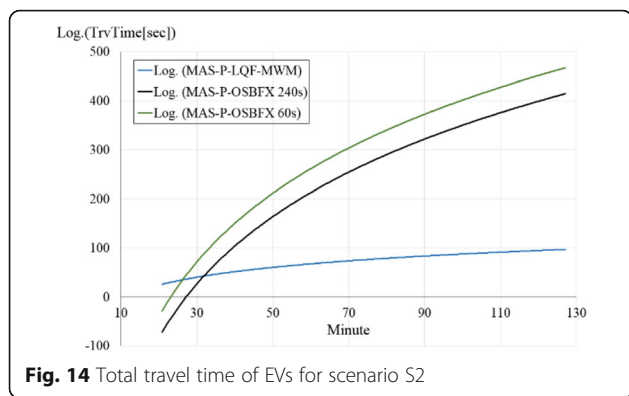


Fig. 13 AVG total delay of EVs for scenario S2



8 Results

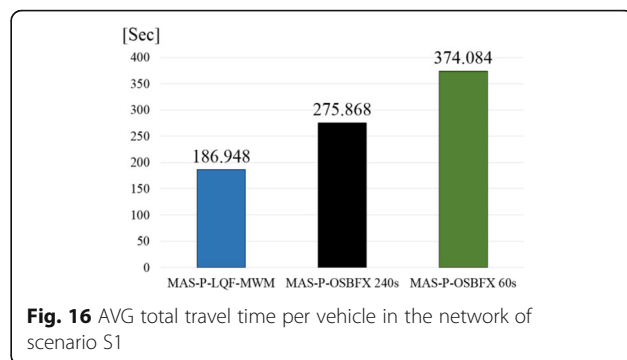
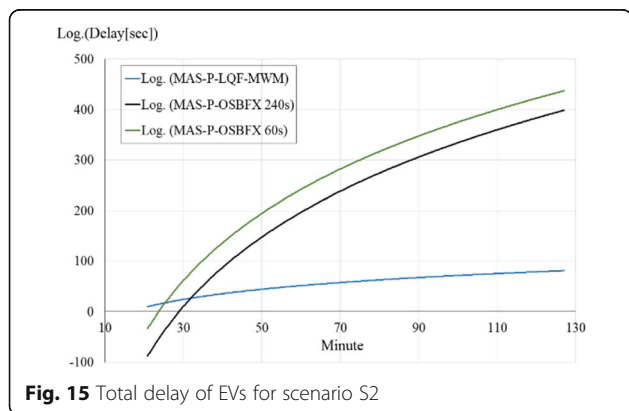
In this section, we evaluate the suggested systems with respect to two criteria. The first criteria relies on the systems capability to manage the crossing of EVs, while the second criteria relies on their capability to maintain a fluid traffic. The results are divided into two main subsections. The first subsection details the results of EVs KPIs, while the second subsection details the results of the network KPIs.

8.1 EV results

This section details the results of EVs KPIs. We evaluate the performance of the suggested systems for all scenarios with regard to seven KPIs, which are the average number of stops of EVs, the EVs average speed of EVs, the average distance traversed of EVs, the average total travel time per EV, the average total delay per EV, the total travel time of all EVs, and the total delay of all EVs.

8.1.1 Scenario S1

In scenario S1 (see Table 6) where the distribution of vehicles are stable and equitable, the EVs have to stop around 2 times until reaching their destinations with



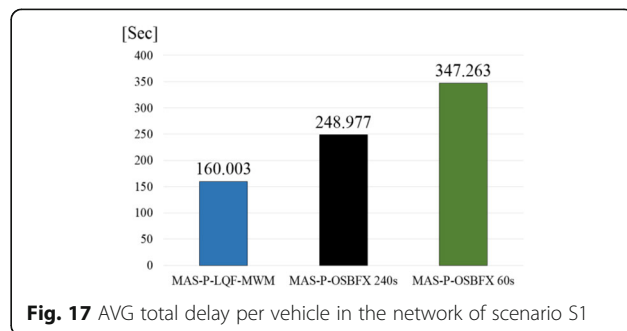
MAS-P-LQF-MWM while they have to stop 7 and 12 times with MAS-P-OSBFX-60s and MAS-P-OSBFX-240 s, respectively. EVs speed is very critical for people lives, MAS-P-LQF-MWM allows a higher average speed which is 30 km/h, while MAS-P-OSBFX-60s and MAS-P-OSBFX-240 s give lower speed, which are 20 km/h and 15 km/h, respectively. The MAS-P-LQF-MWM has an efficient agent communication system that allows to reduce the average distance traversed by 20.6% compared to MAS-P-OSBFX-60s and by 74.07% compared to MAS-P-OSBFX-240 s.

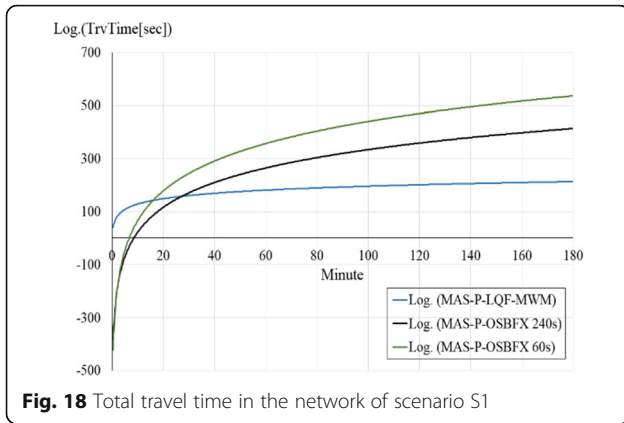
Figures 8 and 9 measure the average travel time and the average delay per EV, respectively. The MAS-P-LQF-MWM gives better performance compared to the MAS-P-OSBFX-60s and the the MAS-P-OSBFX-240 s.

With regard to the total travel time and the total delay of EVs represented in Figs. 10 and 11, the MAS-P-LQF-MWM has a comparable performance with the MAS-P-OSBFX-60s but it still the best performer. The MAS-P-OSBFX-240 s is a poor performer.

8.1.2 Scenario S2

In scenario S2 (see Table 7) where the distribution of vehicles are unstable, the EVs have to stop around 1 times until reaching their destinations with MAS-P-LQF-MWM, while they have to stop 31 and 5 times with MAS-P-OSBFX-60s and MAS-P-OSBFX-240 s, respectively. The MAS-P-LQF-MWM allows higher average

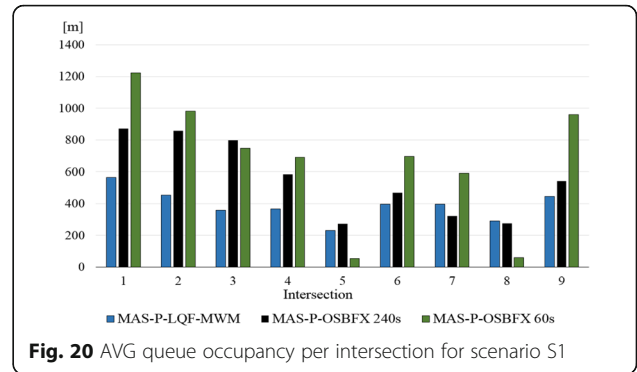
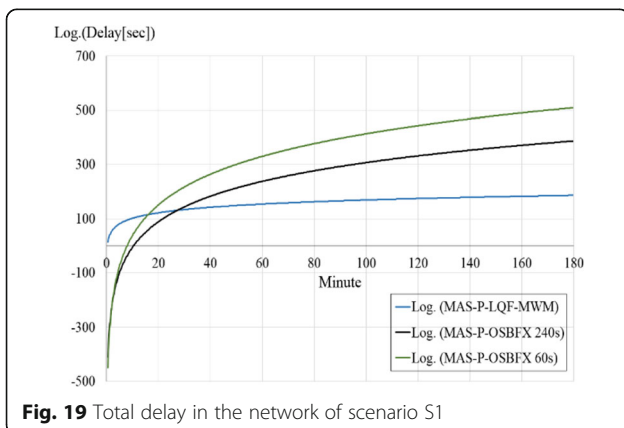




speed (40 km/h), while MAS-P-OSBFX-60s and MAS-P-OSBFX-240 s provide lower speed which are 5 km/h and 11 km/h, respectively. The third KPI which is the distance traversed of EVs shows the efficiency of the suggested MAS-P-LQF-MWM. Indeed, due to the effective communication and collaboration in decision making between agents, the MAS-P-LQF-MWM have reduced the distance traversed by EVs by 82.81% compared to MAS-P-OSBFX-60s and by 53.9% compared to MAS-P-OSBFX-240 s.

It is worth noting through Figs. 12 and 13 that the gap between the MAS-P-LQF-MWM and the other controllers become bigger compared to scenario S1 (see Section 8.1.1). This fact shows the efficient coordination between the agents of the suggested system which coordinate efficiency with the instability of the traffic load in order to reduce the travel time and the delays of EVs. The MAS-P-LQF-MWM is the better performer in scenario S2.

With regard to the total travel time and the total delay of EVs represented in Figs. 14 and 15, MAS-P-LQF-MWM is the best performer, while the MAS-P-OSBFX-60s is the poorest performer.



8.2 Network results

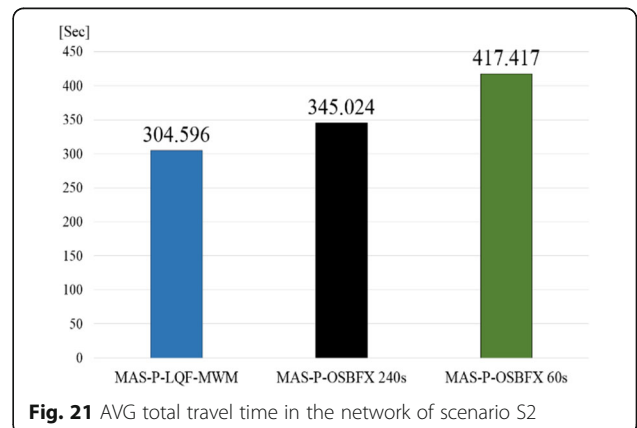
Giving priority to EVs should not damage the fluidity of intersections. In this section, the impact of all algorithms on the network fluidity is investigated.

8.2.1 Scenario S1

Figures 16 and 17 show that the MAS-P-LQF-MWM reduces both travel time and delay in the network. MAS-P-LQF-MWM reduces the total travel time by 47.5% and 50.02% compared to MAS-P-OSBFX-240 s and MAS-P-OSBFX-60s, respectively. Furthermore, MAS-P-LQF-MWM reduces the total delay by 35.7% and 53.92% compared to MAS-P-OSBFX-240 s and MAS-P-OSBFX-60s, respectively.

Figures 18 and 19 show that MAS-P-LQF-MWM achieve lower travel time and delay since the 20th minute until the end of the simulation. MAS-P-OSBFX-60s and MAS-P-OSBFX-240 s continue increasing the lower travel time and delay during the 3 h of simulation.

With regard to the average queue occupancy per intersection for scenario S1 (see Fig. 20), MAS-P-LQF-MWM provides lower average queues compared to all controllers. If we do a pairwise comparison, MAS-P-LQF-MWM reduces the average queue length in 7 intersections of the 9 intersections compared to both controllers, MAS-P-OSBFX-60s and MAS-P-OSBFX-240 s.



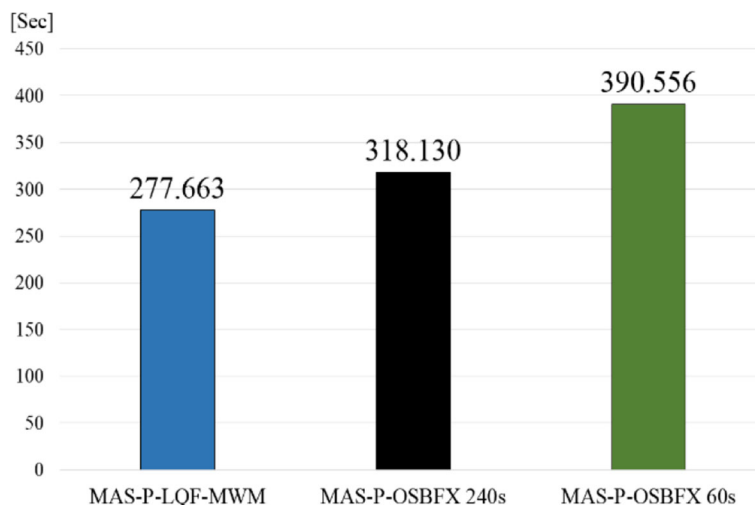


Fig. 22 AVG total delay in the network of scenario S2

8.2.2 Scenario S2

Figures 21 and 22 show that the MAS-P-LQF-MWM react very well to the unstable traffic load and reduces both travel time and delay in the network. The MAS-P-LQF-MWM is the best performer. It reduces the total travel time by 11.7% and 27.02% compared to MAS-P-OSBFX-240 s and MAS-P-OSBFX-60s, respectively. Thus, MAS-P-LQF-MWM reduces the total delay by 12.72% and 28.9% compared to MAS-P-OSBFX-240 s and MAS-P-OSBFX-60s, respectively.

The unstable traffic load don't effect the performance of MAS-P-LQF-MWM which still achieve the lower travel time and delay during the 3 h of simulation (see Figs. 23 and 24). The MAS-P-OSBFX-240 s shows interesting performance and stay close to the MAS-P-LQF-MWM, while the MAS-P-OSBFX-60s

controller continues increasing the lower travel time and delay during the 3 h of simulation.

It is interesting to investigate the queue occupancy per intersection under an unstable traffic load during 3 h of simulation. Figure 25 shows that MAS-P-LQF-MWM achieves lower average queues length in 4 intersection compared to MAS-P-OSBFX-240 s controller, and in 7 approaches compared to the MAS-P-OSBFX-60s controller.

9 Conclusion

Security is a main consideration of any authority in the globe. Thousands of people are dying every day due to Emergency Vehicles (EVs) affected by traffic disturbances. Developing an intelligent Traffic Signal Priority Systems (TSPS) is a challenging problem for which adaptation

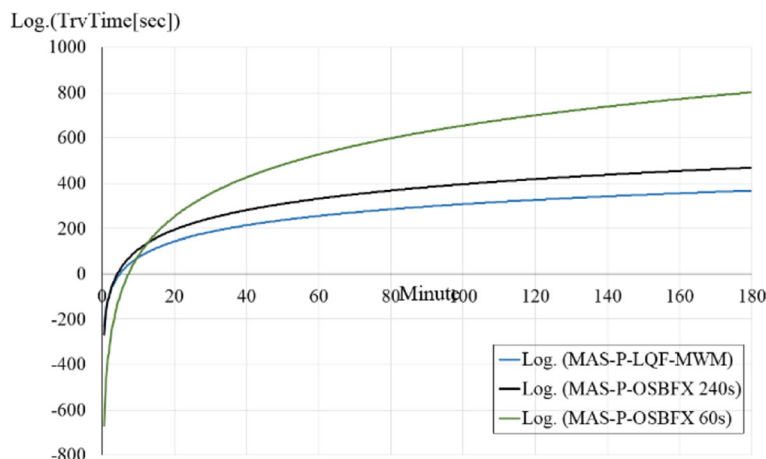
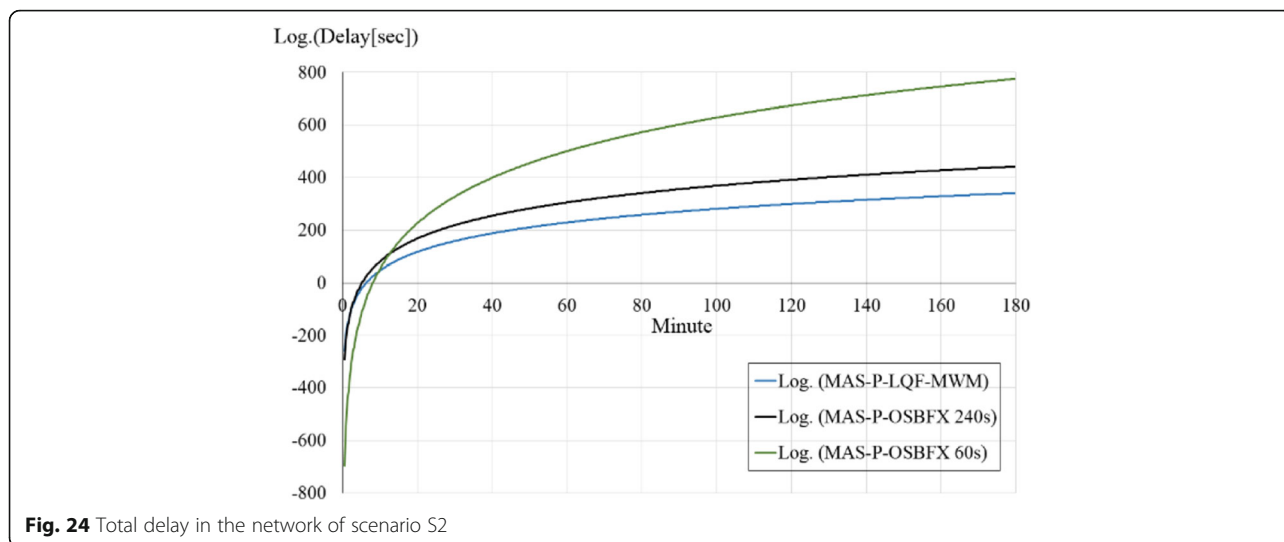


Fig. 23 Total travel time in the network of scenario S2



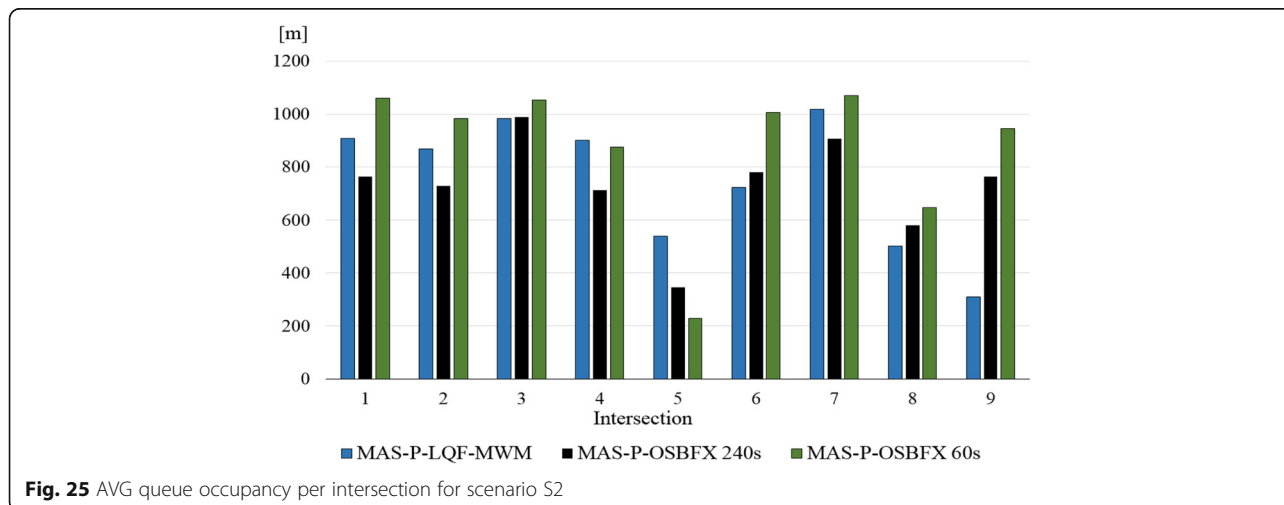
mechanisms need to be developed to deal with EVs in an intelligent way. In this article, we designed an intelligent and adaptive TSPS based on Longest Queue First – Maximal Weight Matching (LQF-MWM) algorithm coupled with preemption mechanisms.

To the best of the authors’ knowledge, this article is the first to integrate preemption mechanisms into the LQF-MWM algorithm (P-LQF-MWM) to develop a TSPS capable to cope with emergencies cases, favour the crossing of EVs, and consider their specifications. Thus, this paper is the first to integrate the suggested P-LQF-MWM into a Multi-Agent System (MAS) to control efficiency disturbed traffic at a network of signalized intersections. The agents architecture provided in this paper is fully heterarchical, and no hierarchy between agents. The EVs guidance and the control of signals are achieved using the P-LQF-MWM and an efficient communication system between agents. The suggested system guarantees six objectives including

minimizing delay time, travel time, queue occupancies, number of stops, distance traversed, and speed.

For the assessment purpose, we customized a distributed and preemptive version of the VISSIM Optimized Stage-Based Fixed-Time algorithm (MAS-P-OSBFX). All controllers have been evaluated in different traffic scenarios. Two types of results are discussed in this paper. The first results evaluate the efficiency of the controllers to deal with EVs, while the second results evaluate the capability of the controllers to maintain a fluid traffic. All results show that the MAS-P-LQF-MWM outperform the MAS-P-OSBFX with regard to several Key Performance Indicators in all scenarios.

This paper provides a detailed benchmark allowing the assessment of any Traffic Signal Control Systems (TSCS) or any Traffic Signal Priority Systems (TSPS). Furthermore, the suggested system provides an explicit knowledge representation that includes the detected emergencies cases and the decisions taken.



This knowledge allows the development of intelligent TSPS using artificial intelligence techniques, such as Case-Based Reasoning and Immune Systems. Due to this knowledge, such systems can easily solve similar emergency cases that could happen in the future.

Acknowledgements

The authors are grateful to Prince Sattam bin Abdulaziz University for funding.

Author contributions

All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Author details

¹Information System Department, Prince Sattam bin Abdulaziz University, 11942 Alkharj, Kingdom of Saudi Arabia. ²University of Tunis, ISG, SMART Lab, 41, Avenue de la Liberté, Bouchoucha, Le Bardo, 2000 Tunis, Tunisia. ³Ecole Centrale Casablanca, Bouskoura Ville Verte, 27182 Casablanca, Morocco. ⁴University "Politehnica" of Bucharest, 313, Splaiul Independentei, 060042 Bucharest, Romania.

Received: 11 January 2018 Accepted: 17 September 2018

Published online: 21 November 2018

References

- Ahmad A, Arshad R, Mahmud SA, Khan GM, Al-Raweshidy HS (2014) Earliest-deadline-based scheduling to reduce urban traffic congestion. *IEEE Trans Intell Transp Syst* 15(4):1510–1526. <https://doi.org/10.1109/ITITS.2014.2300693>
- Araghi S, Khosravi A, Creighton D (2015) A review on computational intelligence methods for controlling traffic signal timing. *Expert Syst Appl* 42(3):1538–1550. <https://doi.org/10.1016/j.eswa.2014.09.003>
- Boillot F, Blosseville JM, Lesort JB, Motyka V, Papageorgiou M, Sellam S (1992) Optimal signal control of urban traffic networks. In: 6th IEE International Conference on Road Traffic Monitoring and Control. IET
- Chakraborty PS, Tiwari A, Sinha PR (2015) Adaptive and optimized emergency vehicle dispatching algorithm for intelligent traffic management system. *Procedia Comput Sci* 57:1384–1393. <https://doi.org/10.1016/j.procs.2015.07.454>
- Chen K-H, Dow C-R, Lin D-J, Yang C-W, Chiang W-C (2008) An NTCIP-based semantic ITS middleware for emergency vehicle preemption. In: 2008 11th International IEEE Conference on Intelligent Transportation Systems. IEEE, pp 363–368. <https://doi.org/10.1109/ITSC.2008.4732608>
- Chen L-W, Sharma P, Tseng Y-C (2013) Dynamic traffic control with fairness and throughput optimization using vehicular communications. *IEEE J Sel Areas Commun* 31(9):504–512. <https://doi.org/10.1109/JSAC.2013.SUP.0513045>
- Darmoul S, Elkosantini S, Louati A, Ben Said L (2017) Multi-agent immune networks to control interrupted flow at signalized intersections. *Transp Res C* 82(Agent technologies):290–313. <https://doi.org/10.1016/j.trc.2017.07.003>
- Doniec A, Mandiau R, Piechowiak S, Espié S (2008) Anticipation based on constraint processing in a multi-agent context. *Auton Agent Multi-Agent Syst* 17(2):339–361. <https://doi.org/10.1007/s10458-008-9048-7>
- FHWA. (2011). Focus on Congestion Relief | Congestion Reduction Toolbox. Retrieved from <https://www.fhwa.dot.gov>
- Gardner DHSB (2009) Review of bus priority at traffic. In: SJ: working program bus committee Retrieved from <http://content.tfl.gov.uk/interaction-of-buses-and-signals-at-road-crossings.pdf>
- Highway Traffic Safety Administration, & Department of Transportation. (2015). Traffic safety facts 2015 motor vehicle crashes: overview
- Houli D, Zhiheng L, Yi Z, Pappis C, Mamdani E, Trabia M et al (2010) Multiobjective reinforcement learning for traffic signal control using vehicular ad hoc network. *EURASIP J Adv Signal Process* 2010(1):724035. <https://doi.org/10.1155/2010/724035>
- Huang, C.-M., Yang, C.-C., Tseng, C.-Y., & Chou, C.-H. (2009). A centralized traffic control mechanism for evacuation of emergency vehicles using the DSRC protocol. In 2009 4th International Symposium on Wireless Pervasive Computing 1–5. IEEE. doi: <https://doi.org/10.1109/ISWPC.2009.4800550>
- Huang Y-S, Shiue J-Y, Luo J (2015) A traffic signal control policy for emergency vehicles preemption using timed petri nets. *IFAC-PapersOnLine* 48(3):2183–2188. <https://doi.org/10.1016/j.ifacol.2015.06.412>
- Li Z, Zhang Y, Tan HZ (2007) An efficient artificial immune network with elite-learning. In: Proceedings - third international conference on natural computation, ICNC 2007, vol 4, pp 213–217. <https://doi.org/10.1109/ICNC.2007.190>
- Little JDC, Kelson MD, Gartner NM (1981) MAXBAND: a program for setting signals on arteries and triangular networks. In: 60th Annual Meeting of the Transportation Research Board. Transportation Research Board, Washington D. C., pp 40–46
- Louati A, Darmoul S, Elkosantini S, Ben Said L (2017a) An artificial immune network to control interrupted flow at a signalized intersection. *Inf Sci* 433–434:70–95 Retrieved from <https://doi.org/10.1016/j.ins.2017.12.033>
- Louati A, Elkosantini S, Darmoul S, Ben Said L (2016) A case-based reasoning system to control traffic at signalized intersections. *IFAC-PapersOnLine* 49(5): 149–154. <https://doi.org/10.1016/j.ifacol.2016.07.105>
- Louati A, Elkosantini S, Darmoul S, Ben Said L (2017b) An immune memory inspired case-based reasoning system to control interrupted flow at a signalized intersection. *Artif Intell Rev*:1–31 Retrieved from <https://doi.org/10.1007/s10462-017-9604-0>
- Marcianò, Alessandro F, Musolino G, Vitetta A (2014) Signal setting optimization on urban road transport networks: the case of emergency evacuation. *Saf Sci* 72:209–220. <https://doi.org/10.1016/J.SSCI.2014.08.005>
- Marsetič R, Šemrov D, Žura M (2014) Road artery traffic light optimization with use of the reinforcement learning. *PROMET - Traffic&Transportation* 26(2):101–108. <https://doi.org/10.7307/ptt.v26i2.1318>
- Martin-Gasulla M, García A, Moreno AT (2016) Benefits of metering signals at roundabouts with unbalanced flow patterns in Spain. *Transp Res Rec* 2585(3):20–28
- Mirchandani P, Head L (2001) A real-time traffic signal control system: architecture, algorithms, and analysis. *Transp Res C Emerg Technol* 9(6):415–432. [https://doi.org/10.1016/S0968-090X\(00\)00047-4](https://doi.org/10.1016/S0968-090X(00)00047-4)
- Mirchandani P, Lucas D (2004) Integrated transit priority and rail/emergency preemption in real-time traffic adaptive signal control. *J Intell Transp Syst* 8(2):101–115. <https://doi.org/10.1080/15472450490437799>
- Negi, P. (2006). Artificial immune system based urban traffic control
- PTV, G. (2015). VISSIM simulation software
- Python Software Foundation. (2017). SPADE 2.3 : Python Package Index
- Qin X, Khan AM (2012) Control strategies of traffic signal timing transition for emergency vehicle preemption. *Transp Res C Emerg Technol* 25:1–17. <https://doi.org/10.1016/j.trc.2012.04.004>
- Robertson DI (1969) "TRANSYT" method for area traffic control. *Traffic Eng Control* 11(6):276–281
- Robertson DI, Bretherton RD (1991) Optimizing networks of traffic signals in real time-the SCOOT method. *IEEE Trans Veh Technol* 40(1):11–15. <https://doi.org/10.1109/25.69966>
- Sharma A, Chaki R, Bhattacharya U (2011) Applications of wireless sensor network in intelligent traffic system: a review. In: 2011 3rd International Conference on Electronics Computer Technology. IEEE, pp 53–57. <https://doi.org/10.1109/ICECTECH.2011.5941955>
- Shibuya S, Yoshida T, Yamashiro Z, Miyawaki M (2000) Fast emergency vehicle preemption systems. *Transp Res Rec* 1739:44–50. <https://doi.org/10.3141/1739-06>
- Tahiyani S, Darbari M, Shukla PK (2013) Soft computing approaches in traffic control systems: a review. *AASRI Procedia* 4:206–211. <https://doi.org/10.1016/j.aasri.2013.10.032>
- Westgate BS, Woodard DB, Matteson DS, Henderson SG (2013) Travel time estimation for ambulances using Bayesian data augmentation 1. *Ann Appl Stat* 7(2):1139–1161. <https://doi.org/10.1214/13-AOAS626>
- Wiedemann R (1974) Simulation des Verkehrsflusses. In: Schriftenreihe Des Instituts Für Verkehrswesen, Heft 8, Universität (TH) Karlsruhe (Seit 2009 KIT – Karlsruher Institut Für Technologie)
- Wu J, Abbas-Turki A, El Moudni A (2011) Cooperative driving: an ant colony system for autonomous intersection management. *Appl Intell* 37(2):207–222. <https://doi.org/10.1007/s10489-011-0322-z>

37. Wunderlich R, Elhanany I, Urbanik T (2007) A stable longest queue first signal scheduling algorithm for an isolated intersection. In: 2007 IEEE International Conference on Vehicular Electronics and Safety. IEEE, pp 1–6. <https://doi.org/10.1109/ICVES.2007.4456393>
38. Wunderlich R, Elhanany I, Urbanik T (2008) A novel signal-scheduling algorithm with quality-of-service provisioning for an isolated intersection. *IEEE Trans Intell Transp Syst* 9(3):536–547. <https://doi.org/10.1109/TITS.2008.928266>
39. Yun I, Best M, “Brian” Park B (2008) Evaluation of transition methods of the 170E and 2070 ATC traffic controllers after emergency vehicle preemption. *J Transp Eng* 134(10):423–431. [https://doi.org/10.1061/\(ASCE\)0733-947X\(2008\)134:10\(423\)](https://doi.org/10.1061/(ASCE)0733-947X(2008)134:10(423))
40. Zamanipour M, Head KL, Feng Y, Khoshmagham S (2016) Efficient priority control model for multimodal traffic signals. *Transp Res Rec* 2557:86–99. <https://doi.org/10.3141/2557-09>
41. Zhang Z, He Q, Gou J, Li X (2016) Performance measure for reliable travel time of emergency vehicles. *Transp Res C Emerg Technol* 65:97–110. <https://doi.org/10.1016/j.trc.2016.01.015>

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- ▶ Convenient online submission
- ▶ Rigorous peer review
- ▶ Open access: articles freely available online
- ▶ High visibility within the field
- ▶ Retaining the copyright to your article

Submit your next manuscript at ▶ [springeropen.com](https://www.springeropen.com)
