Multi-Agent and Fuzzy Inference-Based Framework for Traffic Light Optimization

Abdelouafi Ikidid*, Abdelaziz El Fazziki, Mohammed Sadgal

Cadi Ayyad University, Marrakesh (Morocco)

Received 30 March 2021 | Accepted 30 September 2021 | Published 16 December 2021

Abstract

Despite the fact that agent technologies have widely gained popularity in distributed systems, their potential for advanced management of vehicle traffic has not been sufficiently explored. This paper presents a traffic simulation framework based on agent technology and fuzzy logic. The objective of this framework is to act on the phase layouts represented by its sequences and length to maximize throughput and fluidize traffic at an isolated intersection and for the whole multi-intersection network, through both inter- and intra-intersection collaboration and coordination. The optimizing of signal layouts is done in real time, and it is not only based on local stream factors but also on traffic stream conditions in surrounding intersections. The system profits from agent communication and collaboration as well as coordination features, along with decentralized organization, to decompose the traffic control optimization into subproblems and enable the distributed resolution. Thus, the separate parts can be resolved rapidly by parallel tasking. It also uses fuzzy technology to handle the uncertainty of traffic conditions. An instance of the proposed framework was validated and designed in the ANYLOGIC simulator. Instantiation results and analysis denote that the designed system can significantly develop the efficiency at an individual intersection as well as in the multi-intersection network. It reduces the average travel delay and the time spent in the network compared to multi-agent-based adaptative signal control systems.

Keywords

Agent Technology, Coordination, Communication, Fuzzy Logic, Traffic Signal Regulation.

DOI: 10.9781/ijimai.2021.12.002

I. Introduction

The optimization of signal light control in urban areas is at the forefront of research in the field of Artificial Transportation Systems (ATS). ATS can be implemented by different approaches and technologies. The widely used artificial intelligence techniques for optimizing traffic signals are the Genetic Algorithm, Artificial Neural Network, Fuzzy logic, Multi-Agent System (MAS), Case-Based Reasoning, and Reinforcement Learning (RL). In this paper, we combine agent technology and fuzzy logic to design a cooperative real-time traffic signal optimization system, where the signal control plan is frequently updated to meet the non-stationary traffic state. Agent technologies have been widely accepted as one of the most responsive tools to deal with a wide-reaching distributed system. That's why agent-based systems are well suited for the traffic and transportation domain, since these systems are geographically distributed in a non-stationary environment [1]. Agents can use perceptive data and received information from other agents to achieve their goals. Each agent can cooperate with neighboring agents and adjust his reactions in real time to his surroundings as they change. Therefore, multi-agent technology treats a complicated system in a distributed manner; it splits the complex control system into simple subtasks, therefore allowing parallel and fast decision-making [2].

* Corresponding author.

E-mail address: a.ikidid@gmail.com

With this being considered, the Multi-agent Cooperative Traffic Signal Optimization (MCTSO) is proposed to maximize the signalized intersection throughput and reduce congestion in urban arteries with three contributions: (1) the real-time optimization is introduced to adapt the system in a timely way to the continuously changing conditions and disturbances, supported by online monitoring of the optimum indicators to detect congestion and maintain the system not far off from the suitable operating point as much as possible. (2) Two-stage coordination, including intra-junction coordination, which is enabled to prioritize the higher congested stream, and inter-junction coordination, which is used to generate a fluidized scenario downstream of the congested stream, is used. (3) Distributed collaboration control, splitting the network into sub-areas whose control is easier, is used to allow parallel-tasking. Therefore, the functionality of an MAS will not reside in the agents themselves, but will be ubiquitously distributed to allow the system to perform tasks in parallel, avoiding an additional computational cost [3].

In this article, we propose a distributed and adaptative, as well as online, optimized traffic signal control scheme enabled by a decentralized multi-agent system, where each group of agents represents a signalized intersection control unit, each group coordinates and collaborates with adjacent surrounding groups, and each group achieves local optimization, taking into consideration global network optimization. We use an artificial fuzzy logic algorithm to tackle the fuzzy condition of the road environment. Our proposed MCTSO differs from existing approaches due to agent specialization. The group contains specialist agents for each role, and it is designed and adapted to a specific task, which allows us to improve the agent efficiency and make its role more accurate. Additionally, in MCTSO the combination of two-stages of coordination and collaboration aims to develop a clearer view of the environment, make decisions of common benefit, and avoid local optimization.

The rest of the paper is organized as follows: The second section analyzes and discusses the related works about intelligent traffic signal control and artificial intelligence techniques. The third section gives a global overview of the traffic control problem. The fourth section details the proposed multi-agent system. The fifth section provides the detailed results of the simulation tests carried out on the AnyLogic platform. Finally, we conclude in the sixth section.

II. Related Works

The MAS is rapidly growing as one of the most powerful popular technologies proposed to solve complicated problems in different fields, such as electrical engineering, computer science, electronic commerce, civil engineering, and transportation systems.

In a transportation system and with the diversity of actors involved, agent technology can be used in the various components of the system, such as drivers and vehicles [4] [5], traffic light [6], phases [7], and to handle diverse aspects, e.g., congestion [8], the green transportation system [9], and route guidance[10]. In urban traffic networks, signalized intersections are one of the most important and influential ingredients, and the traffic signal is the most utilized instrument for scheduling and managing traffic flow. In what follows we analyze and discuss succinctly several studies that use a multi-agent system and artificial intelligence techniques to perform intelligent traffic signal control.

Regarding the architecture of multi-agent-based signal control, most approaches usually divided the road network into regions or sub-parts that cover one or more intersections. These sub-parts are controlled by a cell of one or more agents. The organizational structure of agents can be modeled in various ways. The organizational structure determines the interactions, roles, and structures of the agent's community. It can be designed in many forms, such as flat, hierarchical, holonic, teams, and federation [11].

Many studies have reported using a hierarchical scheme to manage the traffic signal. Jin and Ma [12] use reinforcement learning to introduce a hierarchical multi-agent-based control scheme. The agents are categorized as the region agent (RA), intersection agent (IA), and turning movement agent (TA), listed in the order of the hierarchy. Communication and cooperation between agents at equal levels are elevated through the decentralized representation of the framework. Nevertheless, agents at the lower level have to reach an accord between their own goals and those given by the agents on the next level up. Like Jin and Ma, Xu et al. [13] introduced a three-layer optimizing control system that includes intersection controller agents (ICAs), sub-zone controller agents (SZCAs), and network controller agents (NCAs), which represent the lowest, middle, and highest layers, respectively. The interaction takes place across all levels to optimize the signal timing strategy, while coordination is granted by the SZA. Nonetheless, besides the overcharge data at higher levels, the focal decision process might produce a bottleneck in these levels, lengthen the response time, and limit the system's scalability. Flat [14] and holonic [15] structures are also proposed for multi-agent-based traffic signal control. Otherwise, it is widely recognized that there is no specific operating multi-agent architecture that is absolute for all traffic signal control systems; additionally, various operating models can be combined.

Pre-timed signal control cannot adapt to the non-stationary traffic state. It has been a while since interactive system control became a trend in traffic management. The first appearance of adaptative traffic control was in the last decade of the second millennium, with the release of the cycle and offset optimization technique (SCOOT) in the 1980s, the Sydney cooperative adaptive traffic system (SCATS), and the green link determining (GLIDE) system. Thereafter, these adaptative control systems were implemented in many countries to manage traffic control in metropolitan areas, and others have been developed (for a review of the self-adaptive traffic signal control, see [16]).

Recently, more focus has been placed on multi-agent-based systems for urban traffic management [17]. It has been proposed that several transport system problems be solved in a distributed manner. However, disturbance management requires particular abilities that a MAS cannot guarantee alone. Consequently, to create intelligent traffic signal controllers, a MAS integrates various intelligent techniques. For example, many models combined the multi-agent approach with the RL approach to optimize a signal timing plan [18]. The agents employ their ability to communicate with the environment to learn and optimize their decision-making behavior. Foremost among the model-free RL methods, Q-learning (QL) is the model most used by researchers using multi-agent reinforcement learning (MARL) in intelligent traffic light control. A work [19] uses fuzzy Q-learning and agent technologies to develop a traffic lights control framework. Each agent interacts with neighbor agents by getting a reward from each decision. . The control decision is made by using the number of vehicles input to schedule green phase duration. The aim is to maximize the reward and decrease average delay time. El-Tantawy et al. [20] improve the travel time and overall delay using QL and a decentralized junction-based model. The model-free RL can be implemented when dealing with a non-deterministic model of the environment, as it does not require pre-assignment of the environment.

Concurrently, some researchers investigated the potential of fuzzy-logic-based control, which has a rule-based inference system and is based on human reasoning. FL is suitable for handling the control of a single intersection [21] characterized by uncertainty, fuzzy circumstances, inexact data, and typically controlled by rules. Because the MAS has a restricted capability to deal with fuzzy circumstances, the incorporation of an MAS and fuzzy inference can show considerable effectiveness in enhancing signal settings in traffic light control [22] [23].

In these studies, the cooperation mechanism is limited at the inter-junction level, which reduces the local control efficiency in favor of global control. Also, the concentration of fuzzy logic in one level creates an overload at fuzzy components. Our proposed multi-agent control system is a model based on the two levels of coordination and collaboration, local at the intersection and within the surrounding neighbors. Each intersection is represented by a controller group in which the decision is made via two levels of fuzzy logic and coordination with adjacent group controllers.

III. Traffic Control Problem Description

According to the US Census Bureau, metropolitan areas will contain 6.7 billion people [24]. This growing urbanization increases the traffic road demand because of a high number of vehicles seeking to use the road infrastructure. Road traffic in urban areas is a nested phenomenon, on the one hand because of the many contributors that act autonomously and on the other hand because of the uncertainty of the road network. When the number of vehicles on an infrastructure exceeds its capacity, traffic congestion occurs, resulting in slow movements and queues that stretch over time. Therefore, the congestion is a parallel evolutionary anomaly, in both space and time. Consequently, to inexpensively mitigate this anomaly, we can optimize traffic space occupation with an acceptable delay. Signal control is the basic method and an effective one to alleviate congestion as well as to fluidize traffic at the intersection [25]. Optimized signal control can significatively increase infrastructure capacity and reduce travel time [26]. Additionally, it helps to reduce fuel consumption and the emission of air pollutants and improves the health of citizens, too [27].

A. Signalized Intersection Features

A road intersection is a crossing of several roads that contains three functional zones (Fig. 1) managed by a tricolor traffic light; the red queues the vehicles in a storage area, the green gives access to the exit zone through the conflict zone, and the yellow is a transition period from green to red to allow the vehicles to evacuate the conflict zone.

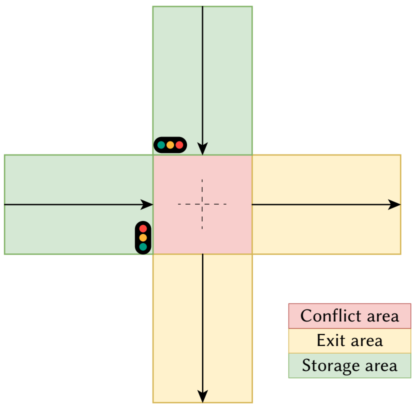


Fig. 1. Functional areas of an intersection of two one-way roads.

B. Intersection Network Modeling

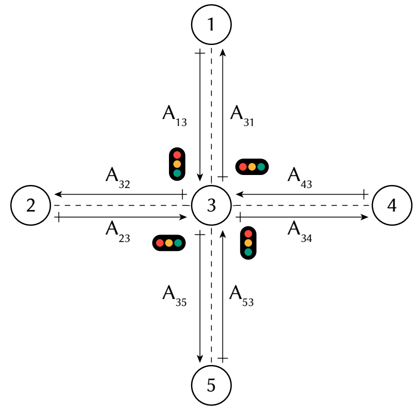
The intersection network is viewed as a disturbed system that is modeled by a strongly connected oriented graph $G = (C, A)$, where (C) is a set of nodes that represent the intersections and (A) is a set of arcs that represent the links that connect these intersections. Each intersection, as a component of the disturbed system, has its own requirements; therefore, it coordinates with its adjacent intersection. Two intersections connected by an arc are considered adjacent. Adjacent intersections cooperate and share their data to achieve a common goal of the system, which is to optimize traffic flow management.

Each arc A_{ij} is bounded by two intersections: i, the initial junction and the arc flow origin, and j, the terminal junction and the arc flow destination. Downstream of an arc is the group of succeeding arcs $succ(A_{ij}) = \{A_{jk}, (i, j, \mathrm{k}) \in C\}$, where the outflows of the arc can be routed. Upstream of an arc is the set of predecessor arcs $pred(A_{ij}) = \{A_{ki},(i, j, \mathrm{k}) \in C\}$ where the inflows of the arc arrive.

An arc is characterized by static information, such as the storage area length and max capacity, and dynamic information, namely the state of the traffic signal at the stop line of the arc (green or red). The concentration T (1) at a given segment is the number of vehicles N (in private vehicle units, PVUs) using this segment at a time t, relative to the segment length.

$$T_{\Delta x}(t) = T(p \to p + \Delta p, t) = \frac{N}{\Delta p} \tag{1}$$

An intersection has a set of incoming arcs $E(i) = \{A_{ji}, (i, j) \in C\}$ and a set of outgoing arcs $S(i) = \{A_{ji}, (i, j) \in C\}$. All intersections are controlled by a signal light, concerning the colors to be used and also their succession or order of appearance. A green phase is a lap of time during which a group of compatible arcs is activated, i.e., the arc flows are allowed to cross the intersection. The cycle is the appearance order of all of the phases, and a traffic control strategy is the scheduling method that defines how the phases participate in the cycle and their layout (length and sequence). Fig. 2(a) shows an intersection with 4 incoming and outgoing arcs and 4 adjacent intersections. Fig. 2(b) gives a representation of a cycle, phase, and the arcs that are activated during each phase.



(a)

Phase		Arc							
		A_{13}	A_{31}	A_{23}	A_{32}	A_{43}	A_{34}	A_{53}	A_{35}
Cycle	$\varphi 1$	1	1	0	1	0	1	0	1
	$\varphi 2$	0	1	1	1	0	1	0	1
	$\varphi 3$	0	1	0	1	1	1	0	1
	$\varphi 4$	0	1	0	1	0	1	1	1

(b)

Fig. 2. Representation of activated and not-activated arcs in a 4-phase traffic light intersection. (a) Intersection with 4 incoming arcs and 4 outgoing arcs; (b) truth table of 4-phases and the arc cycle for each phase.

An intersection is considered congested if it does not manage to evacuate all of the storage areas of the activated arcs after a green phase time; in other words, it is considered congested if the stop time of an incoming arc exceeds the cycle time duration.

IV. Agent Modeling

The organizational design of the urban traffic-responsive control system (UTCS) is spatially and functionally distributed. Each intersection is viewed as a network sub-section and controlled by a community of autonomous, cooperative, and intelligent agents. Commonly, agents are perceived as analyzing at a level with an abstraction upper than components and objects, which makes a MAS suitable for complex and distributed problems.

The proposed MAS has a decentralized architecture with two levels of collaboration. Each signalized intersection is controlled by an intersection control group (ICG), which defines the signal control strategy. This strategy optimizes phase layouts while it is executed to meet the continuously changing surrounding environment, whereas the control of the whole intersection network is fully distributed and is accomplished through the collective and coordination capability of ICGs. In sum, the system goal is achieved with two levels of coordination, which are the following:

- Inter-junction collaboration: This allows coordination between connecting ICGs.
- Intra-junction collaboration: This allows interactions between the agents belonging to the same ICG.

We build our MAS by applying the concept of the model-driven architecture (MDA) [28] to construct our system. We propose to create an increasingly detailed system from the abstract to a concrete concept following a process in five stages as follows:

1. Select the organizational structure of the MAS.
2. Analyze the system requirements.
3. Structure the UTCS into groups of agents.
4. Identify agents and roles.
5. Implement the generic structure of an agent-oriented system in the AnyLogic simulator.

A. The Organizational Structure of the MAS

The selection of the organizational structure is a very essential stage in MAS development and has an impact on the succeeding stages. Various specifications drive the definition of the organizational structure, including the environment characteristics, the architecture of the real-world organization, the ability of the MAS to support the computation and coordination complexity of the scenario, and the necessity of respecting the organizational rules and minimizing the complexity of the design.

Our proposed MAS has a decentralized architecture with two levels of collaboration based on the metamodel AALAADIN [29], which is built on three main notions: agent, group, and role. Fig. 3 shows a diagram of this model.

Agent: The agent is defined as an active entity that communicates and plays a specific role inside its group. The metamodel does not pose any constraint on the internal architecture of agents.

Group: The group is an atomic aggregate of agents sharing services with other groups. Each agent belongs to a group; in our case, the concept of belonging to a group is limited to one group.

Role: The role is an abstract representation of an agent's tasks, function, or activities. Each agent can have multiple roles, and each role is accomplished by an agent group.

We define the organizational structure as a decentralized set of group sharing services.

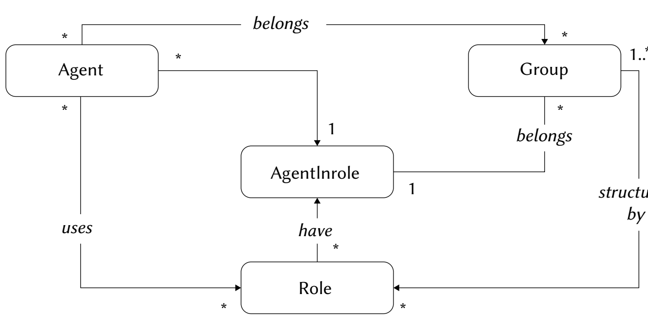


Fig. 3. Organizational structure of the group of agents.

B. Analyzing the System Requirements

The functional architecture of the UTCS includes a set of components. Fig. 4 represents the main components of a regulation system. These components generate an optimized traffic control strategy based on the following scenario:

- The optimization process is initialized after each recurring interval to update the traffic signal control plan.
- The captures are placed at the stop line of the incoming arc to develop a local view of the traffic conditions by observing the storage areas.
- The incoming arcs are monitored to define and update the arc traffic state indicators. These indicators are calculated by observing the local state collected by the captures and considering that of succeeding arcs.
- During the optimizing process, if the degree of saturation in the downstream is intolerable, the upstream indicators are adjusted to slow evacuation and relieve saturation.
- The coordinator provides the traffic condition stat of connecting intersections and shares the local intersection stat.
- The phase managers use the traffic state provided by arc monitors to define the phases' states and request a traffic signal control update.
- The cycle time is prefixed and divided between all intersection phases. Unused lap time will be reallocated to other phases or subtracted from the cycle time.
- The intersection controller updates the control strategy during the progress of the cycle.
- Each arc has a right to green time one and one time only in the cycle, and all the links with at least one queued vehicle at the stop line should have green time.
- The pedestrian phase is outside of the scope of our approach.

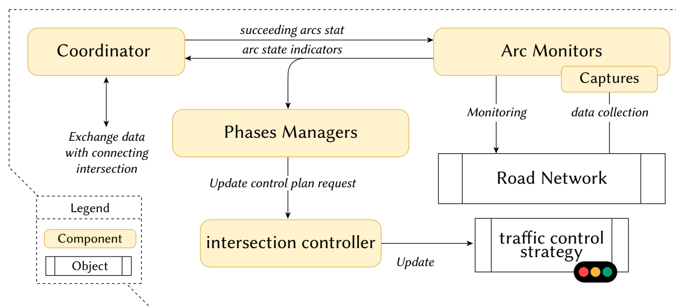


Fig. 4. Components of intersection control.

C. Structure the UTCS Into Groups of Agents

We assume that the structural aspect of a MAS consists of two parts: a classification structure and a role structure (see section D). The classification structure indicates how agent groups are determined and how they interact with each other. This classification is based on the roles of agents and their social interactions. To structure the agent community into groups, we rely on the classic software engineering rule: high cohesion and low coupling. Therefore, agents sharing more roles and goals will be in the same group, and agents that do not share roles, or have few common roles, are placed in two distinct groups.

The representation of a UTCS by a MAS is based on a mapping between the UTCS and MAS that we propose (Table I). Commonly, each intersection is managed by a group control that consists of five main components: an arc monitor for each incoming arc; two phase managers, one for the green phase and the other for the red phase; a coordinator; and an intersection controller.

TABLE I. Alignment of UTCS / MAS

UTCS component	Agent Group
Arc monitor	Intersection Control Group (ICG)
Phase managers	
Coordinator	
intersection controller	

D. Identify Roles and Agents

The MCTSO applied to the signalized intersection network contains a set of ICGs. Each ICG was assigned to an intersection and charged with full control over the incoming streams. Each ICG includes a number of agents classified into 5 types: an ARC agent, which is associated with each incoming arc, two Phase agents (the Active Phase agent (APA) manages the current green phase and the Inactive Phase agent (IPA) manages other phases), a Coordinator agent (CA), and a Decision agent (DA).

1. ARC Agent

Each incoming arc is managed by an agent. The goal of this kind of agent is to monitor the arc storage area in a timely and continuous manner. Arc agents have only a local view of the environment. To minimize the complexity degree of the system, no agent can have a full overview of the network. They use sensors placed at the stop line that cover the whole storage zone to define the arc state, taking into account the outflow streams. Depending on the signal state at the arc stop line, the arc stat is defined by the urgency indicators when the signal is red, which are the stop ratio (SR) and congestion ratio (CR). They are calculated using data collected from sensors and the congestion ratio from downstream (CRd) received from the Coordinator agent. When the signal is green, the arc stat is defined by extend indicators, which are the CR and CRd.

SR (2) represents the waiting time ratio of vehicles in the storage area and is defined as the duration of elapsed red time since the last switch (t_s) divided by the cycle length (c) minus the total yellow signal length (t_y). *CR* (3) is the number of enqueued vehicles in the arc storage zone over the capacity of the arc.

$$SR = \frac{t_s}{c-ty} \tag{2}$$

$$CR = \frac{T_t}{Tmax} \tag{3}$$

where *Tmax* is the maximum concentration of vehicles in the arc; T_t is the concentration at an instant t; t_s is the vehicle stop time on the red signal; c is the cycle length; and ty is the yellow signal length.

To reduce the phase transitions when there is no traffic, arcs waive their green turn by setting the urgency indicators or extend indicators equal to 0 when

- There are no enqueued vehicles in arcs, since the empty arc does not need green time.
- The CRd is equal to or greater than 1, which means that the concentration downstream surpasses or reaches its maximum capacity, since the congested outgoing arc is not able to get more inflow.

This type of agent will be conscious of all of the other intersection agents. It cooperates with the Coordinator agent to define the arc traffic condition state, with the phase agents to propose the suitable phase layout update and with the Decision agent to implement the optimized control strategy.

2. Phase Agents

A phase is seen as a state machine. This automaton has two states: Active and Not Active (Fig. 5). Depending on the states of a phase, we have adopted two agents to manage all phases in an intersection: the APA, which manages the activated phase, and the IPA, which manages the not-activated phases.

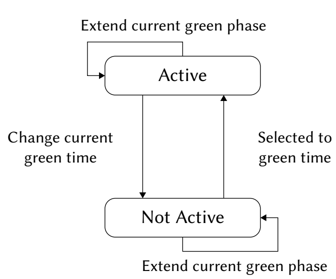


Fig. 5. Phase state.

The goal of the APA is to maximize the green time allocated to the arcs that make up the current green phase, while the IPA's goal is to reduce the stop time of the other phases. The phase agents compete; each agent seeks to extend its active time and otherwise limit the time of the other phase agents.

a) IPA

The IPA controls the phase sequences. It selects a phase from all phases, except the current phase and the already activated phase in the cycle, to be a candidate for the next green time. This agent examined the urgency degree of approved phases by evaluating arc urgency indicators provided by arc agents; a phase is represented by the most urgent arc. The phase urgency degree is obtained by the fuzzy process after the verification of the max/min constraint.

Selection of candidate phase:

The candidate phase is calculated through the collaboration and coordination with the **Arc** and **Coordinator** agents. The **IPA** is the controller of phase scheduling and sequence. It proposes an advisable phase order for the current traffic state. The flowchart of the candidate phase selection process is presented in Fig. 6 and consists of six steps:

Step 1. The **IPA** starts the phase selection process by creating a *collaboration-group* and initializes it to the list of all intersection arc agents classified in the phase set (one arc agent may belong to many phases).

Step 2. The **IPA** sends a request to the *collaboration-group members* to inform them that the phase selection process has been started and orders them to begin calculating the arcs' urgency indicators.

Step 3. Each arc agent of the *collaboration-group* calculates its urgency indicators.

Step 4. The **IPA** receives all responses and calculates the urgency degree of each phase using a *fuzzy selection mechanism.* The highest urgency phase will be selected and suggested to receive green time. The selected phase and its urgency degree value will be sent to the **Decision** agent.

Step 5. If the suggested phase gets the green time, the **IPA** removes it from the *collaboration-group* list; it also removes their arc agent if they did not belong to other phases of the current *collaboration-group members.*

Step 6. The phase selector waits a predetermined time (min. red time) and returns to **step 2** while the *collaboration-group* is not empty.

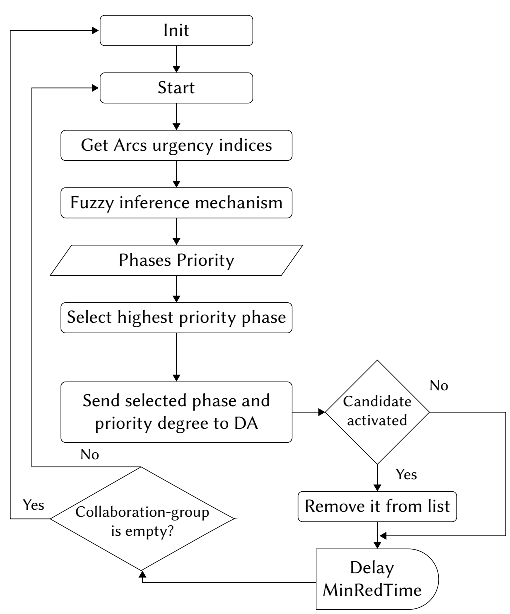


Fig. 6. Flowchart of the candidate phase selection process.

Fuzzy selection mechanism:

The IPA combines the inputs CR, CRd, and SR to create the urgency degree output of the candidate phases. The outputs of the phase agents are used as input in the DA process to make the final decision.

The membership functions SR, CR, and CRd are standardized. According to this, there are four membership functions, including Small (S), Medium (M), Large (L), and Very Large (VL), for these inputs. The linguistic variables as well as the membership functions are shown in Fig. 7. The technique used in all of the defuzzification process is the Center of Gravity (COG) method.

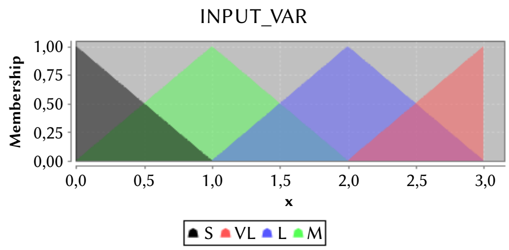


Fig. 7. Membership functions of inputs.

To define the most appropriate phase for green time, a set of rules is defined for the phase selection process. Fig. 8 shows several rules that were used to perform the system simulation. The "AND" operation is performed by using the max t-norm (Łukasiewicz).

```
RULE 11 : IF CR IS M AND CRd IS S AND SR IS S THEN UrgencyDegree IS S ;
RULE 12 : IF CR IS M AND CRd IS S AND SR IS M THEN UrgencyDegree IS M ;
RULE 13 : IF CR IS M AND CRd IS S AND SR IS L THEN UrgencyDegree IS L ;
RULE 14 : IF CR IS M AND CRd IS M AND SR IS S THEN UrgencyDegree IS S ;
RULE 15 : IF CR IS M AND CRd IS M AND SR IS M THEN UrgencyDegree IS M ;
RULE 16 : IF CR IS M AND CRd IS M AND SR IS L THEN UrgencyDegree IS L ;
RULE 17 : IF CR IS M AND CRd IS L AND SR IS S THEN UrgencyDegree IS S ;
```

Fig. 8. Fuzzy rules of phase urgency degree determination process.

b) APA

This type of agent is charged with managing the activated phase. Its goal is to maintain if possible the green time for the current phase until it has evacuated all of its enqueued vehicles. Using the extended indicator provided by active arc agents, it calculates the phase extended degree. The extended degree indicates the extended green time need level. To define the extension need degree, the **APA** collaborates with the set of arc agents involved in the current green phase.

The phase extender process is executed after each 1/3 of allocated green time, which the same as the min. red time. This will synchronize the two-parallel process of phase agents. The APA starts the extend process by sending a request message to the agents managing the active arcs demanding the extend indicators.

The extended indicators of all active arcs will be passed into the fuzzy mechanism to determine the phase extend degree and send it to the DA to request an extension. As mentioned previously, the input CR and CRd are standardized. An example of the rules used to define the extended degree is shown in Fig. 9**.**

```
RULE 3 : IF CR IS S AND CRd IS M THEN ExtendDegree IS PNo;
RULE 4 : IF CR IS S AND CRd IS L THEN ExtendDegree IS PNo;
RULE 5 : IF CR IS M AND CRd IS S THEN ExtendDegree IS PYes;
RULE 6 : IF CR IS M AND CRd IS M THEN ExtendDegree IS Maybe;
RULE 7 : IF CR IS M AND CRd IS L THEN ExtendDegree IS PNo;
RULE 8 : IF CR IS L AND CRd IS S THEN ExtendDegree IS Yes;
```

Fig. 9. Fuzzy rules of phase extend degree determination process.

3. Coordinator Agent

The objective of the CA is to coordinate with the connecting control group. It represents the communication interface agent of the ICG and plays a mediator role in all external communications. It exchanges the state of incoming arcs with the adjacent CA group member. It takes the succeeding arcs stat request from the local arc agents and contacts the CAs of the appropriate groups to get the requested data and response to the request. For its part, it provides the local arc stat to other groups. The CA controls all of the interaction flow with the ICG outside the environment, and it assures coordination and collaboration with others.

4. Decision Agent

The DA is the agent axis of our architecture; it decides to extend the active phase or switch to the selected phase. The decision is made in a collaborative way to avoid local optimization. The DA receives simultaneous requests from phase agents and then decides via fuzzy inference to either extend the current phase or to switch to the candidate phase. This agent sends the final decision to the phases and arc agents in real time.

The DA starts the decision process by checking the parameters of the phases to evaluate if the max. elapsed time of red and green time is reached. Then, it uses a fuzzy mechanism to make the decision and informs the phase and arc agents. Fig. 10 shows the decision-making process.

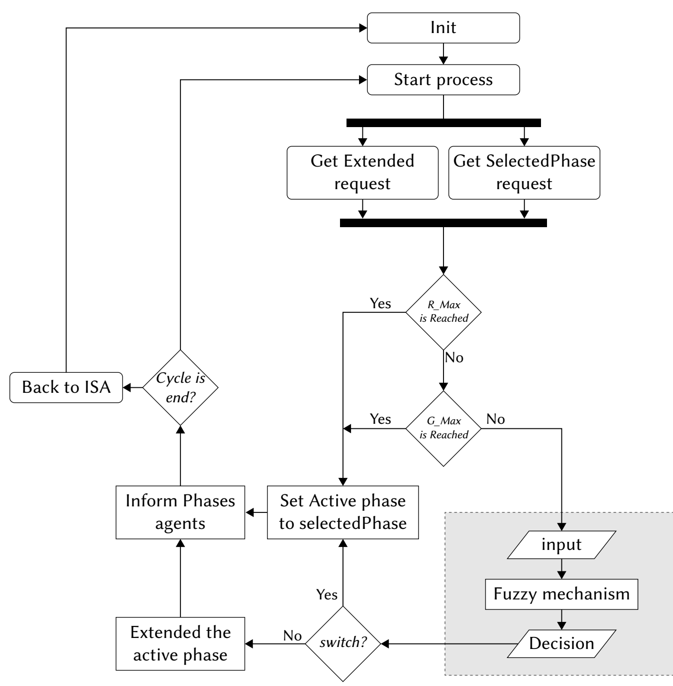


Fig. 10. Fuzzy decision mechanism.

The DA uses the phase urgency degree and extend degree provided by the phase agents to make the final decision. As in Fig. 11, there are five membership functions, including Zero (Z) Low (L), Medium (M), High (H), and Very High (VH), for the phase urgency degree. For the extended degree, there are five membership functions, including No (N), Perhaps No (PNo), Maybe Yes (MYes), Perhaps Yes (PYes), and Yes (Y). Finally, there are only two membership functions for the decision to switch to a candidate phase: No and Yes.

The decision-making process is based on a set of fuzzy rules. Fig. 12 shows an example of these rules.

Fig. 11. Membership functions.

```
RULE 13 : IF ExtendDegree IS Mybe AND UrgencyDegree IS Mybe THEN Decision IS  No ;
RULE 14 : IF ExtendDegree IS Mybe AND UrgencyDegree IS PYes THEN Decision IS Yes ;
RULE 15 : IF ExtendDegree IS Mybe AND UrgencyDegree IS  Yes  THEN Decision IS Yes ;
RULE 16 : IF ExtendDegree IS PYes AND UrgencyDegree IS  No   THEN Decision IS  No ;
RULE 17 : IF ExtendDegree IS PYes AND UrgencyDegree IS PNo  THEN Decision IS  No ;
RULE 18 : IF ExtendDegree IS PYes AND UrgencyDegree IS Mybe THEN Decision IS  No ;
```

Fig. 12. Fuzzy rules of the decision process.

The interaction of different agents in the designed MCTSO is illustrated in Fig. 13. All agents have a communication model to perceive their environment and handle the exchanged data flow within the agent society. A common Agent Communication Language (ACL) has been used to fulfill the communication model goals.

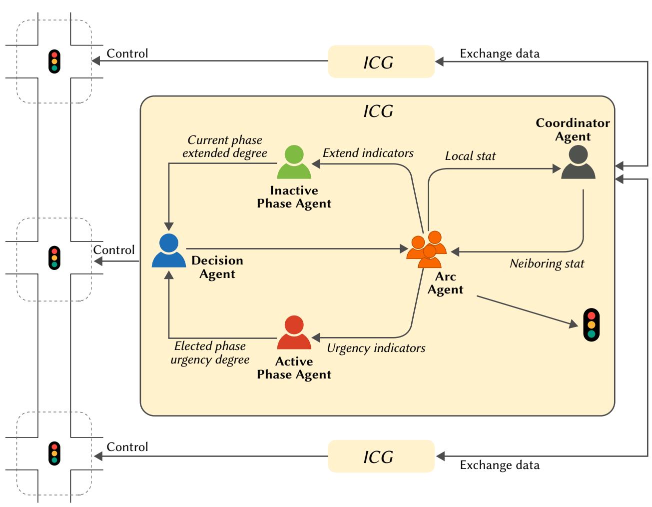


Fig. 13. The architecture of the MCTSO.

V. Experimental Results and Performance Analysis

The performance of the proposed system is validated and designed in the ANYLOGIC simulator, which is used to handle both the agent modeling and traffic simulation using a virtual road network that is shown in Fig. 14. It consists of 9 intersections controlled by 9 ICGs.

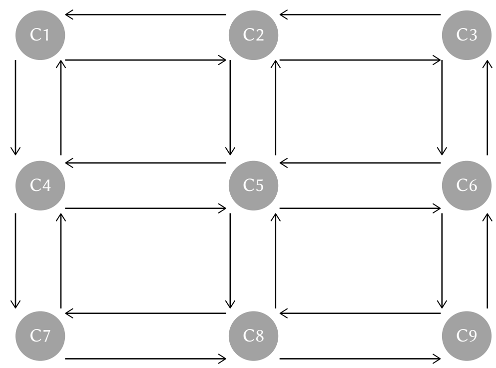


Fig. 14. Road network.

The fuzzy inference mechanism has been programmed in the JAVA language, using jFuzzyLogic, an open-source Fuzzy Logic library, and an FCL language implementation, which offers a fully functional and complete implementation of a fuzzy inference [30]. To link jFuzzyLogic to AnyLogic, we add the jFuzzyLogic library to Java external libraries in AnyLogic [31].

Fig. 15 describes the procedure of setting up a simulation for the MCTSC system. First, the road network is extracted in image format. AnyLogic is then used to convert the image into a simulation network. After obtaining the simulation network, vehicle mobility is generated using an origin-destination matrix. The arrival rate is adjusted to simulate the different scenarios of traffic demand. Then, the agent-based-modeling approach of AnyLogic is used to implement agents, and jFuzzyLogic library to handle the fuzzy decisions.

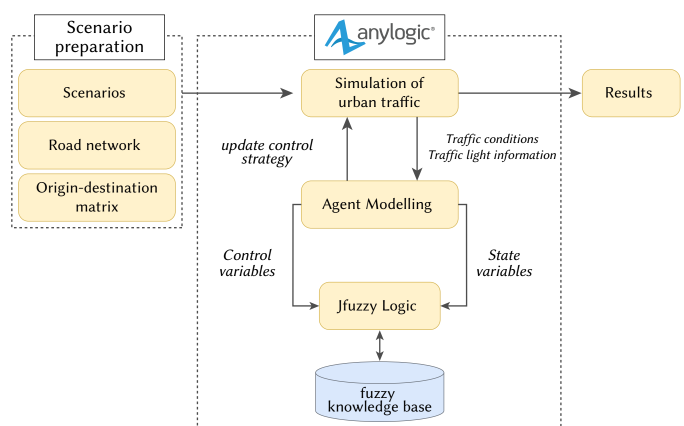


Fig. 15. The procedure of simulation.

A. Experimental Setup

In the study area, the frequency of entry at the source point is adjusted to simulate the varying traffic demand from the peak hour to the slack period. During the simulation, new vehicles are generated with an origin-destination matrix (OD). The OD simulates requests for network uses and represents possible situations of urban traffic conditions. The evaluation is carried out at both local intersections, to evaluate the local optimization, and for the whole network, to evaluate a global optimization.

Arcs are monitored up to the storage area length. We assume that the arc storage area can be varied proportionally to the lane number and the length of the given link. For links longer than 400 m, we monitor the point 150 m from the stop line, or 30 PVU, with 5 m as the typical car length, and all of the links in the other cases.

B. Results and Analysis

We assess the performance of MCTSO by referring to two control methods, namely the adaptative traffic signal optimization and adaptative multi-agent traffic signals control proposed by [32]:

- Adaptative traffic signal optimization (ATSO): standard version of the proposed MCTSO without agents.
- Adaptative multi-agent traffic signals control (AMTSC): represents a control traffic signals method based on multi-agent systems to control the traffic signals. The agents are organized in holonic architecture and a holonic Q-learning method is adopted to learn signals timing in two holarchical levels.

Since a feasible approach should smoothly deal with different traffic conditions, all control systems are tested on similar conditions and under 3 different scenarios: the first scenario allows the assessment of the performance of methods under low traffic demand, with 18000 PVU/hour as arrival rate. The second scenario describes medium traffic demand and represents a moderate congestion situation, with an arrival rate of 27000 PVU/hour. The third scenario provides results for high traffic demand with an arrival rate of 36000 PVU/hour. Each method is run for 180 minutes in each scenario case. Each case is repeated for 50 iterations to increase the reliability of the collected results.

In this study, the vehicle travel time and travel speed are chosen to build up an overview of the general performance of the control methods. The travel time represents the time between the departure of the vehicle from the origin point and the arrival at the destination point. Such criteria will provide us with the optimization level of our approach, and it includes the average stop time and network throughput indices.

Table II depicts the average vehicle's travel time and speed for each assessment scenario. It particularly shows that MCTSO allows the fastest travel time under all scenarios, and by consequence improves the number of vehicles that can use the network and reach their destinations compared to the other control methods. The results show that all control methods have acceptable performance under the first scenario, due to the low level of traffic flow. Increasing the arrival rate causes the congestion to get worse; the ATSO method presents an unacceptable travel time and speed. Thus, ATSO is not proficient with medium and higher congestion levels. At the higher level of congestion, our proposed method optimizes both travel time and speed, and it reduces these criteria compared to other controllers. We notice that in low traffic demand the average travel time attained by our proposed framework is 16,71% lower than that in ATSO, 6,82% than the AMTSC. While in medium traffic demand the corresponding improvement of the proposed framework is 20,36% and 11,08, respectively. The improvements become more important in high traffic demand and attain 37,65% compared to ATSO and 22,05% compared to AMTSC. As to the average travel speed, we observed that the MCTSO provided 8,26% according to the ATSO model and 4,82% improvement according to AMTSC in low traffic conditions, while this improvement rises 20,91% and 13,21% according to ATSO and AMTSC respectively in normal traffic density. In a heavy density scenario, the proposed model has better speed performance about 26,57% compared to the ATSO model and 16,76% compared to AMTSC. Furthermore, the standard deviation of the vehicle's travel time and travel speed of the proposed approach is lower than that in the other control methods. A high standard deviation means that there is a large amount of variability among the data, while a low standard deviation means that the data is less spread, thus very reliable. Consequently, the proposed approach is more reliable.

In addition, the ANOVA two factor with replication test yields a p-value of ≈ 0 (3,39E-40 for travel time and 2,18E-97 for travel speed) that is much smaller than the 0,05 level of alpha significance, meaning that the changes in used control methods had statistically a significant impact on the travel time in different traffic demand.

The reduction in travel time is due to the reduction in the set of key performances and by consequence, in a set of intersection indices. Fig. 16 summarizes the intersection metrics' key performances. The measurements are first locally aggregated in each intersection and at each time over evaluation scenarios; then, the performance average and other indices are calculated. The measurements show that MCTSO outperforms other controllers' methods in almost all metrics. Unfortunately, other methods failed to optimize green time management to mitigate traffic conditions.

Regarding the indices in Fig. 16, our proposed controller gives more throughput cars with less green time in all scenarios. This is due to optimizing the splitting of green time over all phases, reutilizing unused green time in phases with no cars in the storage zone. Also, the reduction in the average red time minimizes the travel time. The results also show a reduction in the number of cycles per intersection, which means that the system is suspended due to an empty storage zone in all intersection arcs. This augments the performance of the control system and makes evacuating the surrounding intersection more likely.

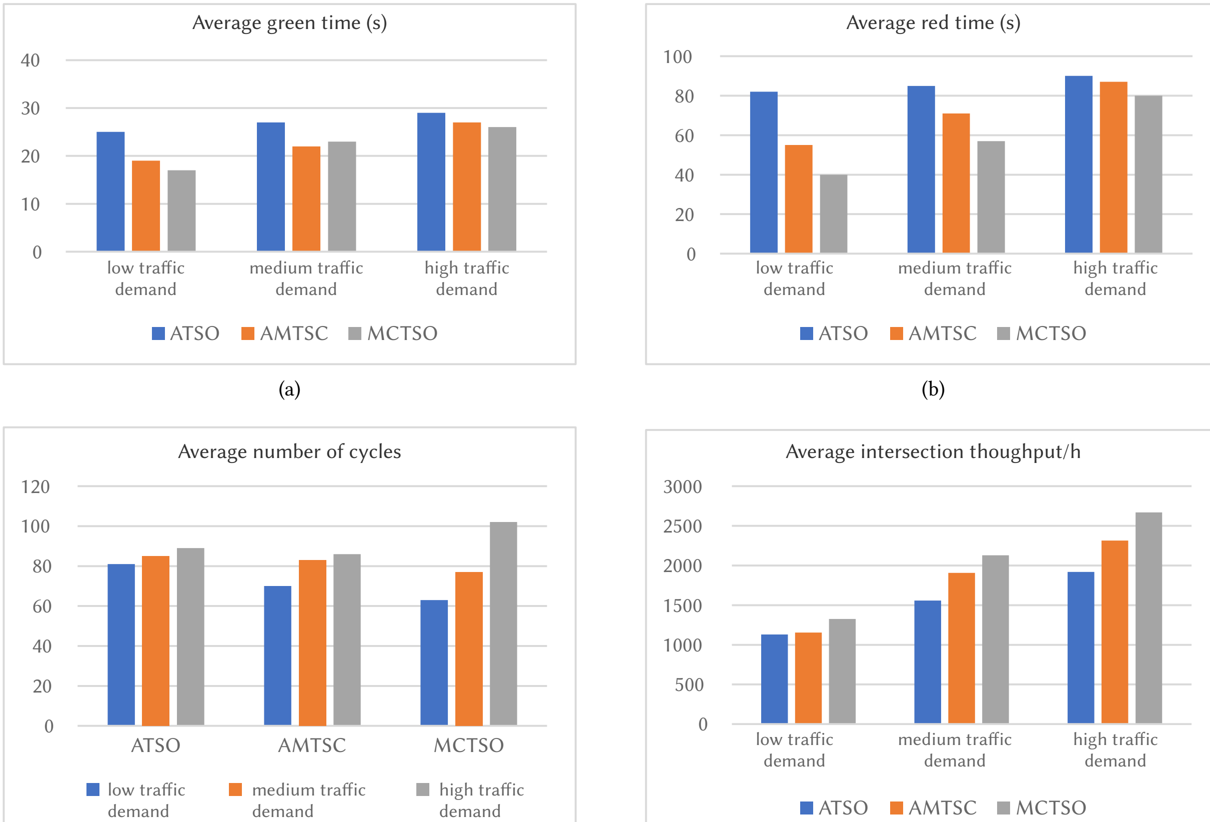
The results show that the proposed system is a practical approach and works smoothly with different traffic conditions.

VI. Conclusions

In this paper, a Multi-agent Cooperative Traffic Signal Optimization (MCTSO) is proposed to reduce congestion on urban roads by optimizing traffic light control with three contributions. First, the MCTSC interactive system involves real-time optimization. Second, there are two levels of coordination, the inter-junction and intra-junction, to avoid local optimization and build a control strategy

TABLE II. Percentage Improvement in Travel Time and Speed Over the Other Control Methods

	low traffic demand			medium traffic demand			high traffic demand		
Parameters	**ATSO**	**AMTSC**	**MCTSO**	**ATSO**	**AMTSC**	**MCTSO**	**ATSO**	**AMTSC**	**MCTSO**
Avg. travel time (s)	76,00	69,56	65,12	96,00	88,60	79,76	130,96	116,12	95,14
Standard deviation of delay	7,84	6,78	4,07	7,62	6,77	4,55	7,33	6,57	4,55
improvement travel time (%)	16,71%	6,82%	N/A	20,36%	11,08%	N/A	37,65%	22,05%	N/A
Avg. travel speed (km/h)	38,76	40,03	41,96	29,13	31,11	35,22	22,51	24,4	28,49
Standard deviation of delay	8,34	7,76	5,08	7,86	7,49	4,15	8,61	6,26	4,22
improvement travel time (%)	8,26%	4,82%	N/A	20,91%	13,21%	N/A	26,57%	16,76%	N/A



(a) (b) (c) (d)

Fig. 16. The intersection metrics' key performances. The first chart shows the average green time assigned to active phases, the second chart shows the average red time assigned to inactive phases, the third chart shows the average number of cycles, and the fourth chart shows the average intersection throughput. All these measurements had been taken for each method and under different scenarios.

that takes into consideration all connecting intersections. Third, distributed control restricts the cooperative scope in neighbors and allows the parallel control.

The proposed system can handle a large multi-intersection network with many alterations in the road infrastructure, and hence facilitates extensibility. The system also increases the robustness and throughput of the network, as shown in the simulation executed in the AnyLogic simulator. The performance of the proposed approach has been compared to the same approach without agents and another adaptative multi-agent optimization. In these performance comparisons, the average travel time and speed were selected as signaling performance criteria. In low traffic demand scenarios, the proposed Multi-agent Cooperative Traffic Signal Optimization model provided 16,71%–6,82% improvement in average travel time and 8,26%–4,42% improvement in average travel speed compared to both adaptative traffic signal optimization and adaptative multi-agent traffic signals control respectively. These improvement values become more important when traffic demand increases and the traffic congestion goes worst, and they are respectively up to 37,65%–22,05% for average travel time and to 26,57%–16,76% for average travel speed compared to adaptative traffic signal optimization and adaptative multi-agent traffic signals control models in high traffic demand. Both local and network performance keys are investigated based on the computational experiments in different traffic condition scenarios. The proposed agent-based optimization shows a better result and can adapt smoothly with different traffic demands. It can significantly optimize performance keys such as travel time, stop time, intersection throughput, and so on.

In the future, the framework shall be further extended to other traffic control fields. For traffic signal control, one extension of this approach is to include priority vehicles and add the special management of priority links. Meanwhile, it is necessary to develop the intelligent optimization approach for operations concerning large uncertainties in the road network, such as disturbances and emergencies.

References

[1] B. Chen and H. H. Cheng, "A review of the applications of agent technology in traffic and transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 2, pp. 485–497, 2010, doi: 10.1109/TITS.2010.2048313.

[2] M. R. Evans and D. S. Elston, "Agent-Based Modeling and Simulation for Transportation, VASTO: Evolutionary Agent System for Transportation Outlook," pp. 1–88, 2013.

[3] de la P. F. Román J.A., Rodríguez S., *Highlights of Practical Applications of Scalable Multi-Agent Systems. The PAAMS Collection*, vol. 616. Cham: Springer International Publishing, 2016.

[4] K. Małecki, "A computer simulation of traffic flow with on-street parking and drivers' behaviour based on cellular automata and a multi-agent system," *Journal of Computational Science*, vol. 28, pp. 32–42, 2018, doi: 10.1016/j.jocs.2018.07.005.

[5] H. Hamidi and A. Kamankesh, "An Approach to Intelligent Traffic Management System Using a Multi-agent System," *International Journal of Intelligent Transportation Systems Research*, vol. 16, no. 2, pp. 112–124, 2018, doi: 10.1007/s13177-017-0142-6.

[6] T. Anagnostopoulos, C. Luo, J. Ramson, K. Ntalianis, V. Kostakos, and C. Skourlas, "A multi-agent system for distributed smartphone sensing cycling in smart cities," *Journal of Systems and Information Technology*,

vol. 22, no. 4, pp. 119–134, 2020, doi: 10.1108/JSIT-12-2018-0158.

[7] Z. Zhang, J. Yang, and H. Zha, "Integrating independent and centralized multi-agent reinforcement learning for traffic signal network optimization," *Proceedings of the International Joint Conference on Autonomous Agents and Multiagent Systems, AAMAS*, vol. 2020-May, pp. 2083–2085, 2020.

[8] M. C. Ho, J. M. Y. Lim, K. L. Soon, and C. Y. Chong, "An improved pheromone-based vehicle rerouting system to reduce traffic congestion," *Applied Soft Computing Journal*, vol. 84, p. 105702, 2019, doi: 10.1016/j.asoc.2019.105702.

[9] K. L. Soon, J. M. Y. Lim, and R. Parthiban, "Coordinated Traffic Light Control in Cooperative Green Vehicle Routing for Pheromone-based Multi-Agent Systems," *Applied Soft Computing Journal*, vol. 81, p. 105486, 2019, doi: 10.1016/j.asoc.2019.105486.

[10] A. Eydi, S. Panahi, and I. N. Kamalabadi, "User-based Vehicle Route Guidance in Urban Networks Based on Intelligent Multi Agents Systems and the ANT-Q Algorithm," *International Journal of Transportation Engineering*, vol. 4, no. 3, pp. 147–161, 2016.

[11] A. Dorri, S. S. Kanhere, and R. Jurdak, "Multi-Agent Systems: A Survey," *IEEE Access*, vol. 6, pp. 28573–28593, 2018, doi: 10.1109/ACCESS.2018.2831228.

[12] J. Jin and X. Ma, "Hierarchical multi-agent control of traffic lights based on collective learning," *Engineering Applications of Artificial Intelligence*, vol. 68, no. January, pp. 236–248, 2018, doi: 10.1016/j.engappai.2017.10.013.

[13] M. Xu, K. An, L. H. Vu, Z. Ye, J. Feng, and E. Chen, "Optimizing multi-agent based urban traffic signal control system," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 23, no. 4, pp. 357–369, 2018, doi: 10.1080/15472450.2018.1501273.

[14] S. Darmoul, S. Elkosantini, A. Louati, and L. Ben Said, "Multi-agent immune networks to control interrupted flow at signalized intersections," *Transportation Research Part C: Emerging Technologies*, vol. 82, pp. 290–313, 2017, doi: 10.1016/j.trc.2017.07.003.

[15] I. H. Tchappi *et al.*, "A critical review of the use of holonic paradigm in traffic and transportation systems," *Engineering Applications of Artificial Intelligence*, vol. 90, no. March 2019, p. 103503, 2020, doi: 10.1016/j.engappai.2020.103503.

[16] Y. Wang, X. Yang, H. Liang, and Y. Liu, "A review of the self-adaptive traffic signal control system based on future traffic environment," *Journal of Advanced Transportation*, vol. 2018, 2018, doi: 10.1155/2018/1096123.

[17] M. Vasirani, F. Klügl, E. Camponogara, and H. Hattori, "Special issue on intelligent agents in traffic and transportation," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 19, no. 1, pp. 1–2, 2015, doi: 10.1080/15472450.2013.856719.

[18] P. Mannion, J. Duggan, and E. Howley, "An Experimental Review of Reinforcement Learning Algorithms for Adaptive Traffic Signal Control," *Autonomic Road Transport Support Systems*, pp. 47–66, 2016, doi: 10.1007/978-3-319-25808-9.

[19] A. Daeichian and A. Haghani, "Fuzzy Q-Learning-Based Multi-agent System for Intelligent Traffic Control by a Game Theory Approach," *Arabian Journal for Science and Engineering*, vol. 43, no. 6, pp. 3241–3247, 2018, doi: 10.1007/s13369-017-3018-9.

[20] S. El-Tantawy, B. Abdulhai, and H. Abdelgawad, "Multiagent reinforcement learning for integrated network of adaptive traffic signal controllers (marlin-atsc): Methodology and large-scale application on downtown toronto," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 3, pp. 1140–1150, 2013, doi: 10.1109/TITS.2013.2255286.

[21] M. Collotta, L. Lo Bello, and G. Pau, "A novel approach for dynamic traffic lights management based on Wireless Sensor Networks and multiple fuzzy logic controllers," *Expert Systems with Applications*, vol. 42, no. 13, pp. 5403–5415, 2015, doi: 10.1016/j.eswa.2015.02.011.

[22] A. Latif and P. Megantoro, "Traffic Light Regulatory System Based on Fuzzy Algorithm Using Microcontroller," *Journal of Physics: Conference Series*, vol. 1464, no. 1, 2020, doi: 10.1088/1742-6596/1464/1/012034.

[23] Y. Bi, D. Srinivasan, X. Lu, Z. Sun, and W. Zeng, "Type-2 fuzzy multi-intersection traffic signal control with differential evolution optimization," *Expert Systems with Applications*, vol. 41, no. 16, pp. 7338–7349, 2014, doi: 10.1016/j.eswa.2014.06.022.

[24] "World Population Projections - Worldometer." [Online]. Available: https://www.worldometers.info/world-population/world-population-projections/. [Accessed: 05-Feb-2021].

[25] M. Xu, Z. Ye, H. Sun, and W. Wang, "Optimization model for transit signal priority under conflicting priority requests," *Transportation Research Record*, vol. 2539, pp. 140–148, 2016, doi: 10.3141/2539-16.

[26] X. Li and J. Q. Sun, "Signal Multiobjective Optimization for Urban Traffic Network," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 11, pp. 3529–3537, 2018, doi: 10.1109/TITS.2017.2787103.

[27] A. C. Olivera, J. M. García-Nieto, and E. Alba, "Reducing vehicle emissions and fuel consumption in the city by using particle swarm optimization," *Applied Intelligence*, vol. 42, no. 3, pp. 389–405, 2015, doi: 10.1007/s10489-014-0604-3.

[28] M. Kempa and Z. A. Mann, "Model Driven Architecture," *Informatik-Spektrum*, vol. 28, no. 4, pp. 298–302, Aug. 2005, doi: 10.1007/s00287-005-0505-2.

[29] J. Ferber and O. Gutknecht, "A meta-model for the analysis and design of organizations in multi-agent systems," *Proceedings - International Conference on Multi Agent Systems, ICMAS 1998*, pp. 128–135, 1998, doi: 10.1109/ICMAS.1998.699041.

[30] P. Cingolani and J. Alcalá-Fdez, "jFuzzyLogic: A Java Library to Design Fuzzy Logic Controllers According to the Standard for Fuzzy Control Programming," *International Journal of Computational Intelligence Systems*, vol. 6, no. SUPPL1, pp. 61–75, 2013, doi: 10.1080/18756891.2013.818190.

[31] "AnyLogic: Simulation Modeling Software Tools & Solutions for Business." [Online]. Available: https://www.anylogic.com/. [Accessed: 28-Mar-2021].

[32] M. Abdoos, N. Mozayani, and A. L. C. Bazzan, "Holonic multi-agent system for traffic signals control," *Engineering Applications of Artificial Intelligence*, vol. 26, no. 5–6, pp. 1575–1587, 2013, doi: 10.1016/j.engappai.2013.01.007.

Abdelouafi Ikidid

He is a PhD candidate at the Computer Science Department of Cadi Ayyad University in Marrakesh, Morocco. He received his Master's degree in Information Systems Engineering from the same university in 2016. His research interests are in software engineering, focusing on multi-agent systems and artificial Intelligence.

Abdelaziz El Fazziki

He received the M.S. degree from the University of Nancy, France, in 1985, and the Ph.D. degree in computer science from the Cadi Ayyad University in 2002. He is a Professor of computer science at Cadi Ayyad University, where he has been since 1985. He is the author of over 50 papers on software engineering. His research interests are in software engineering and focusing on information system development

Mohammed Sadgal

He received the Ph.D. degree in computer science from the University of Lyon in 1989, and the Ph.D. degree in computer science from Cadi Ayyad University in 2005. From 1985 to 1987, he was an Associate Researcher with Lyon I, France. He is currently a Professor with Cadi Ayyad University, Marrakesh, Morocco. His research interests include computer vision, artificial intelligence, and multi-agent systems.