Software Defined Network-based Control System for an Efficient Traffic Management for Emergency Situations in Smart Cities

Albert Rego[1], Laura Garcia[1], Sandra Sendra[1,2], Jaime Lloret[1]

[1]Universitat Politècnica de València, Valencia, Spain

[2]Dept. of Signal Theory, Telematics and Communications Department (TSTC), Universidad de Granada, Granada, Spain

alremae@teleco.upv.es, laugarg2@teleco.upv.es, ssendra@ugr.es, jlloret@dcom.upv.es

Abstract: Smart cities provide new application based on Internet of Things (IoT) technology. Moreover, Software Defined Networks (SDN) offer the possibility of controlling the network based on applications requirements. One of the main problems that arise when an emergency happens is minimizing the delay time in emergency resource forwarding so as to reduce both human and material damages. In this paper, a new control system based on the integration of SDN and IoT in Smart City environments is proposed. This control system actuates when an emergency happens and modifies dynamically the routes of normal and emergency urban traffic in order to reduce the time that the emergency resources need to get to the emergency area. The architecture is based on a set of IoT networks composed by traffic lights, traffic cameras and an algorithm. The algorithm controls the request of resources and the modification of routes in order to ease the movement of emergency service units. Afterwards, the proposal is tested by emulating a Smart City as a SDN-utilizing Mininet. The experiments show that the delay of the emergency traffic improves in a 33% when the algorithm is running. Moreover, the energy consumed by the IoT nodes is modeled and the obtained results display that it increases linearly with the number of nodes, therefore, the proposal is scalable.

Keywords: Software defined networks (SDN); internet of things (IoT); smart city; emergency situations; quick evacuation; traffic; energy consumption; cloud; Network performance; Jitter; Delay

1. Introduction

At present, smart cities are deployed in many countries. Most of the largest cities of the most developed countries have implemented or are implementing smart city technologies in order to improve the efficiency of the management of their city. Manchester, Malta, Hong Kong, Shanghai, Taipei, San Francisco and Vancouver are some examples of places where smart city technologies have been employed [1]. The functionalities that are part of a smart city are varied. Related to governance, there are services such as e-govern, digitalizing the voting system or improving transparency and accountability. Moreover, aspects such as smart education, smart healthcare or smart environment are considered part of the functionalities of a smart city. Other aspects are related to the overall management of the city introducing technologies to improve the efficiency on energy management [2], public transport and urban traffic management, waste management and water distribution management. Particularly, urban traffic management has been a topic of great interest to both city councils and researchers as problems such as congestion and waiting times, which derived from poor traffic management, have a great impact on the daily life of the citizens. Although accident response has been addressed by many researchers [3], large scale incidents and emergency situations remain fairly underdeveloped.

Fires, accidents, natural disasters or terrorist attacks can affect any place on the planet. Especially, cities may be a challenge when evacuating an area in an efficient way. It is important to consider that human behavior is a critical aspect when performing evacuations. Stress and anxiety can affect drivers inducing them to make rush decisions. However, human behavior in emergency situations can be rational and successful evacuations can be done when providing a set of actions to be followed in such situations [4]. Technology can be employed to guide drivers to a successful evacuation in order to ensure their safety and a faster and more efficient intervention from security forces and medical staff.

Edge of Things (EoT) is a new technology that can be employed for deploying smart cities. It can provide better real-time data analysis, lower latency, higher scalability, lower operational cost and better Quality of Service (QoS) than other technologies applied in smart city services [5]. Meanwhile, Internet of Things (IoT) has been considerably employed for smart city applications. Urban traffic management is no different with the deployment of intelligent traffic signaling. Intelligent traffic lights allow controlling vehicle traffic following the guidelines defined by a control center or an algorithm [6]. Events like congestion, accidents, an emergency vehicle passing by or the evacuation of a zone in danger can be triggers that mark the beginning of a particular course of action. However, although varied technologies such as RFID (Radio Frequency Identification) [7], Vehicular ad hoc networks (VANET) [8] or WSN (Wireless Sensor Networks) [9] have been employed for this purpose, leading edge technologies such as SDN (Software Defined Networks) are yet to be utilized in the context of urban traffic management.

SDN is currently being employed in a wide variety of contexts. It is employed with artificial intelligence [10], for throughput allocation [11], video streaming or mobile networks [12] among other uses. In the context of smart cities, it is starting to be utilized for smart homes and IoT information management [13]. However, it is yet to be a common technology in smart city applications. Employing SDN in smart cities enables to control the network utilizing software in order to communicate and give orders to the IoT nodes. Moreover, the network infrastructure can be used as an interface between cloud services and IoT nodes. By doing so, it is possible to have a centralized control of the network and the system allowing to process events and perform changes in a faster way. Deploying SDN in smart cities and, particularly, in urban traffic management systems, could improve the performance of the system and facilitate the creation of new smart cities and smart applications to improve people’s life and to increase safety in roads and emergency situations.

Considering the benefits of employing SDN on smart cities and urban traffic management, and the lack of other SDN solutions for smart cities, it is extremely important to advance in the research of this solutions. In this paper, we present an SDN-based architecture for urban traffic monitoring in emergency situations in smart city environments. The scale of the emergency situations can vary from small to bigger ones depending on the magnitude of the event. An algorithm to manage emergency events is proposed as well. Furthermore, a priority system is employed in order to forward the available resources. Our architecture allows to modify the routes of the vehicles by giving instructions to the intelligent traffic lights deployed throughout the city. These traffic lights are able to coordinate among each other and modify the duration of their green, yellow and red lights. Other types of traffic signs could act as well in order to forward the traffic toward the modified routes. As a result, we are able to divert urban traffic to the chosen direction and facilitate the intervention of the emergency response team.

The rest of the paper is organized as follows. Section 2 comprises the related work. The proposal is presented in Section 3. Section 4 depicts the energy model of our system. The results of the experiments performed on the operation of the system are provided in Section 5. Finally, the conclusion and future work are presented in Section 6.

2. Related work

Urban traffic management has been a topic that has gathered attention due to its impact on people’s life and its relation to smart cities. In this section, the related work on this topic is presented.

A survey on urban traffic management and congestion avoidance is performed in [14] by Kapileswar Nellore et al. They identify six key factors in urban traffic management, being bottlenecks, traffic incidents, inclement weather, work area, rare events and poor signal timings. Moreover, they consider connectivity and coverage, communication and energy cost, congestion, traffic incident notification and coordination and implementation as the most significant challenges of urban traffic management systems based on WSN. Another analysis on urban traffic management systems is done by Christina Diakaki et al. in [15]. They discuss a variety of Vehicle automation and Communication systems (VACs). They say that VACs are employed for reducing congestion, reducing travel times, improving safety, determining fuel consumption, evaluating the environment and quality of life (QoL). Moreover, they classify VACS considering direct traffic flow implications and distinguishing between urban and motorway traffic.

Urban traffic modelling is considered as a way of acquiring knowledge before the implementation of a traffic management system. Mannan Temani et al. present in [16] an intelligent traffic model that employs LabView, NI Vision Assistant and Vision Acquisition tools for image processing. The system can be applied for over-speeding, casualties, fuel prices, traffic management or corruption. The information on the condition of the roads is displayed on a screen located inside the vehicle. Furthermore, the system is able to activate an alert when the speed-limit is surpassed and when pedestrians crossing the street are detected. A dynamic traffic assignment (DTA) model for emissions and fuel consumption applications is designed in [17] by Xuesong Zhou et al. It provides representations for both microscopic and macroscopic traffic streams. It employs kinematic waves and linear car estimations based on Newell's simplified models. MOVES Lite is utilized for emission estimation and the mesoscopic simulation framework called DATALite is employed for determining the impact on emission, fuel consumption and traffic dynamics of distinct traffic management schemes. Finally, Md. Abdus Samad Kamal et al. propose in [18] a system to improve traffic flow based on predictive control. The system is able to regulate intervehicle distance and improves braking. They employ propagation jamming waves to model vehicular traffic behavior and traffic congestion. The results of the numerical simulation show an improvement of congestion and traffic flow by controlling one vehicle.

Cameras are often employed as a source of information. Luca Calderoni et al. present in [19] a smart city urban traffic management system called city kernel. It is divided in subsystems according to different sensor networks. They employ smart cameras that transmit information over the power line. The system is able to count vehicles, classify vehicles, detect vehicle speed, detect traffic congestion and monitor 8 lanes simultaneously. A system that utilizes life video to determine real-time traffic density is proposed in [20] by Anurag Kanugo et al. The information form cameras deployed at traffic junctions is evaluated employing image processing. Furthermore, it employs an algorithm to switch traffic lights in accordance to traffic density. As a result, it reduces traffic congestion and the overall number of accidents. The system is also able to reduce fuel consumption, waiting time and to improve safety.

Urban traffic management implementation proposals utilize different techniques. Zhiguang Cao et al. present in [21] a vehicle traffic management framework that employs traffic light control and dynamic vehicle rerouting in order to reduce congestion. The framework is based on digital pheromones that are placed over the route. The infrastructure agents collect said pheromones and perform a real-time evaluation of the conditions of the roads. Furthermore, it predicts congestion and decides alternative routes for the vehicles that have not entered the congested area. Results show better performance and robustness compared to other frameworks. An ant colony-based road traffic management system is proposed by Habib M. Kammoun et al. in [22]. It allows real-time road traffic distribution. Simulations on a multiagent platform are performed considering time, fluidity and adaptivity of the traffic. They were done employing the TurtleKit and MadKit tools. Results show improved traffic quality and reduction of travel time. Alexander Artikis et al. develop in [23] a system that applies heterogeneous stream processing and crowdsourcing to urban traffic management. Both fixed and mobile sensors are deployed on intersections and public transport vehicles in order to detect congestion trends. Crowdsourcing is employed to handle sensor disagreement and traffic modelling provides information in low-coverage areas. Jetendra Joshi et al. design in [24] a traffic management service called TMaaS. It employs VANET and Cloud Computing Services. A hardware prototype was created in order to perform measures such as the speed of the vehicle, its location, lane changing and accident detection. Traffic monitoring systems of different areas can access the collected information form a central cloud. Simulations were performed employing the software Simulation of Urban Mobility (SUMO) and Network Simulator (NS2) obtaining a 75% improve in efficiency. Lastly, a V2V (Vehicle-to-Vehicle) based cooperative technique called CoTEC (COperative Traffic congestion detECtion) is proposed by R. Bauza et al. in [25]. Its purpose is to detect traffic congestion and its performance is simulated employing the software called iTETRIS. The results show more than a 90% of congestion detection and a reduction of 88% of communication overhead.

Successful urban traffic management has been performed by many researchers. They employ varied technologies and mostly focus on reducing traffic congestion and waiting times. However, SDN has not been employed for this purpose. In this paper, we present a SDN-based urban management system for traffic flow management in emergency situations employing intelligent traffic nodes.

3. Scenario and proposed architecture

This section presents a typical scenario and the IoT architecture proposed in this work. The section also includes the packet exchange for both the case when an alarm is detected and the case when the alarm is finished. Finally, the employed structure of the messages is described.

3.1 Architecture and Algorithm

IoT networks are composed by nodes that provide data to the network in order to make intelligent and efficient decisions. On the other hand, a SDN is composed by virtual devices, specifically Openflow Switches, which play the role of Cluster Head (CH). In our scenario (See Fig. 1), the traffic elements such as cameras and traffic lights compose each IoT network while each OpenFlow switch is managed by an SDN controller that can make decisions about the behavior of the nodes on the IoT networks. The SDN network is in charge of gathering data from all IoT networks and distributes this information along the rest of the OpenFlow switches. Additionally, the SDN controller communicates with a cloud service which is capable of requiring actions from the controller in order to make changes in the SDN network and in the IoT networks.

As Fig. 1 shows, the entire city is a smart city where the control plane is directed by the SDN network and the data plane is composed by the IoT networks. When an emergency event is registered, the SDN controller receives data from the OpenFlow switches and the SDN controller sends requests for available resources throughout all the networks, as well as the location (by using area IDs) of each IoT network and the possibility of fast reaching the place where the emergency was detected. At that moment, the flows are opened to allow communication between the IoT nodes that have to collaborate in order to divert urban traffic and make decisions so as to obtain the fastest paths for the emergency services.

In order to ease the access of emergency services to the place where the emergency was detected, the SDN network modifies the behavior of the traffic lights changing, for example, the duration of the intervals between sequences of green / amber / red lights for vehicles, the time pedestrians spend crossing or the diversion of vehicles to certain directions. Traffic lights and traffic panels can be employed to modify the routes for vehicles and disperse them from the place of the emergency. In this way, it is possible to coordinate traffic lights that initially were not and, after varying the routes or conditions, this cooperation is needed for better management of the emergency situation. The pseudocode of the algorithm is presented in Algorithm 1. First, the type of the emergency is obtained from the emergency ID, if the emergency is labeled as a high risk emergency, additional security resources are asked. Then, the shortest route are calculated for each resource, and taking into account that some resources can share parts of the route to the emergency area. Finally, the SDN flows of the CH are modified, the CH are informed about the emergency and the behavior of the IoT nodes are communicated.

Algorithm 1. Alert treatment

```
Given: Alert_ID, Alert_Area, Resources

Alert.Type = Get_Type(Alert_ID)
If Alert.Type == HIGH_RISK do
        Local_Resources = Ask for Resources to the Local Security service
        Resources = Resources U Local_Resources
(CH_Set, Flows, Behaviors) = Calculate_Routes(Resources, Alert_Area)
Send the OpenFlow Modification Messages(CH_Set, Flows)
Send Emergency_Notification(CH)
```

```
For each IoT Node n in CH_Set
        Send FL Behavior(Behaviors)
```

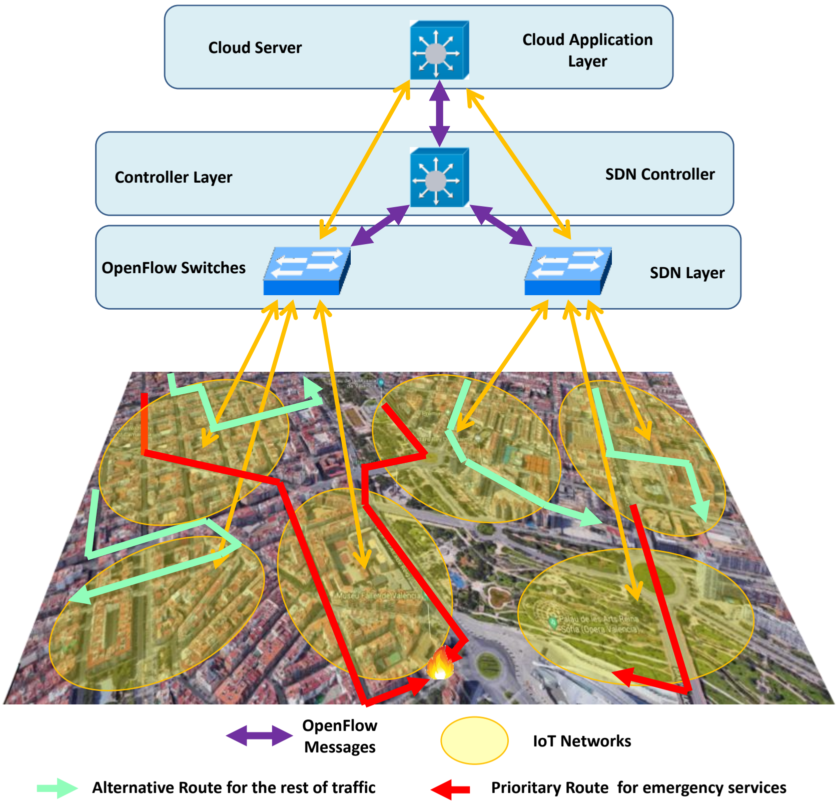


Fig. 1. Scenario and proposed architecture

In order to completely understand the algorithm, the messages exchanged are described in the next subsection.

3.2 Messages

In this subsection, the structure of the exchanged messages is detailed. The exchange of messages when an alert occurs is displayed in Fig. 2. Fig. 3 represents the flow of the messages when the emergency has been handled.

Fig.2 describes the process of communication when the emergency service notifies an emergency alert. It sends the message to the SDN controller that read the data to know if the emergency has been classified as a high risk emergency. If it is a high risk emergency, the SDN controller asks for resources to the local security service located in the city. The service provides local security resources and notifies there are via Resource Allocation messages. After that, the algorithm determines the routes for the normal traffic and for every emergency resource available. By using OpenFlow Flow Modification messages the controller changes the communication between the IoT networks. Then, informs the CH of the affected areas that the behavior of the node have to change due to an emergency, using the FL Emergency Behavior message. The FL Behavior set the behavior of an individual IoT node.

In a similar way, Fig. 3 describes the communication established to set the behavior of the nodes back to the usual way of operating. The message flow is similar to the described in Fig.2. Instead of asking for resources, the SDN controller informs that the resources has been released because of the end of the emergency. Then, modifies the flow table again in every CH and informs them that they have to change the behavior of their nodes to the one that has been previously used. That is performed via FL Normal Behavior message.

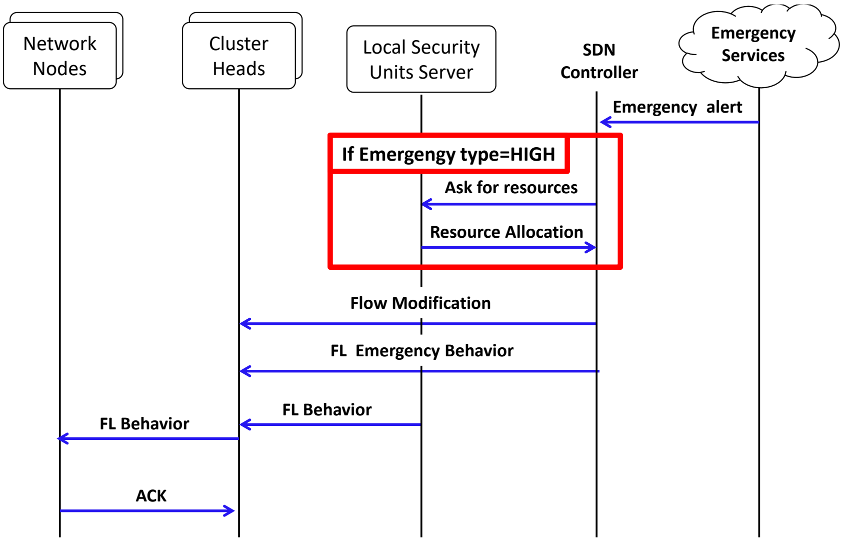


Fig. 2. Message exchange when alarm is detected

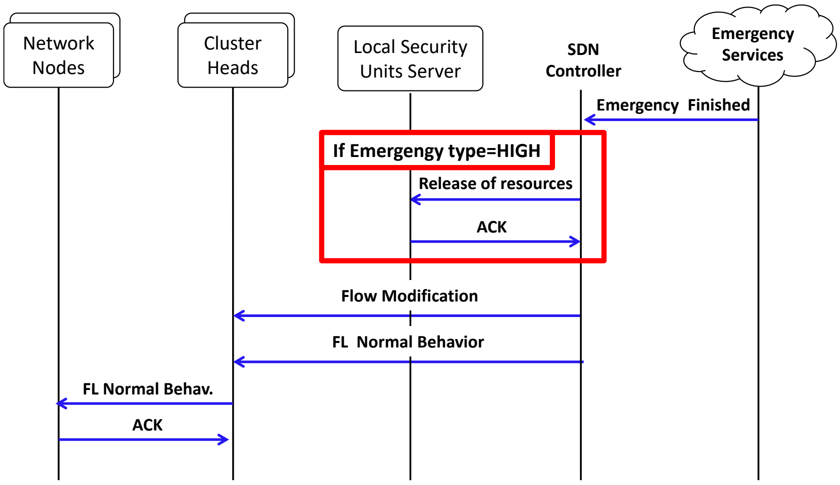


Fig. 3. Message exchange when alarm is finished

The structure of the messages is used in Section 4 for calculating the energy consumed in the transmission during an operation.

In Fig. 4 the messages using in the algorithm are described. All the messages used in this proposal follow the structure TLV (type, length, value). As it is showed in Fig. 4, the field Type is composed by 4 bits, and can describe 16 different types of messages. Then, the field Length (1 Byte) is used to inform about the size of the message. Finally, the field Value contains the data that the applications need in order to work properly.

The messages used in the protocol are:

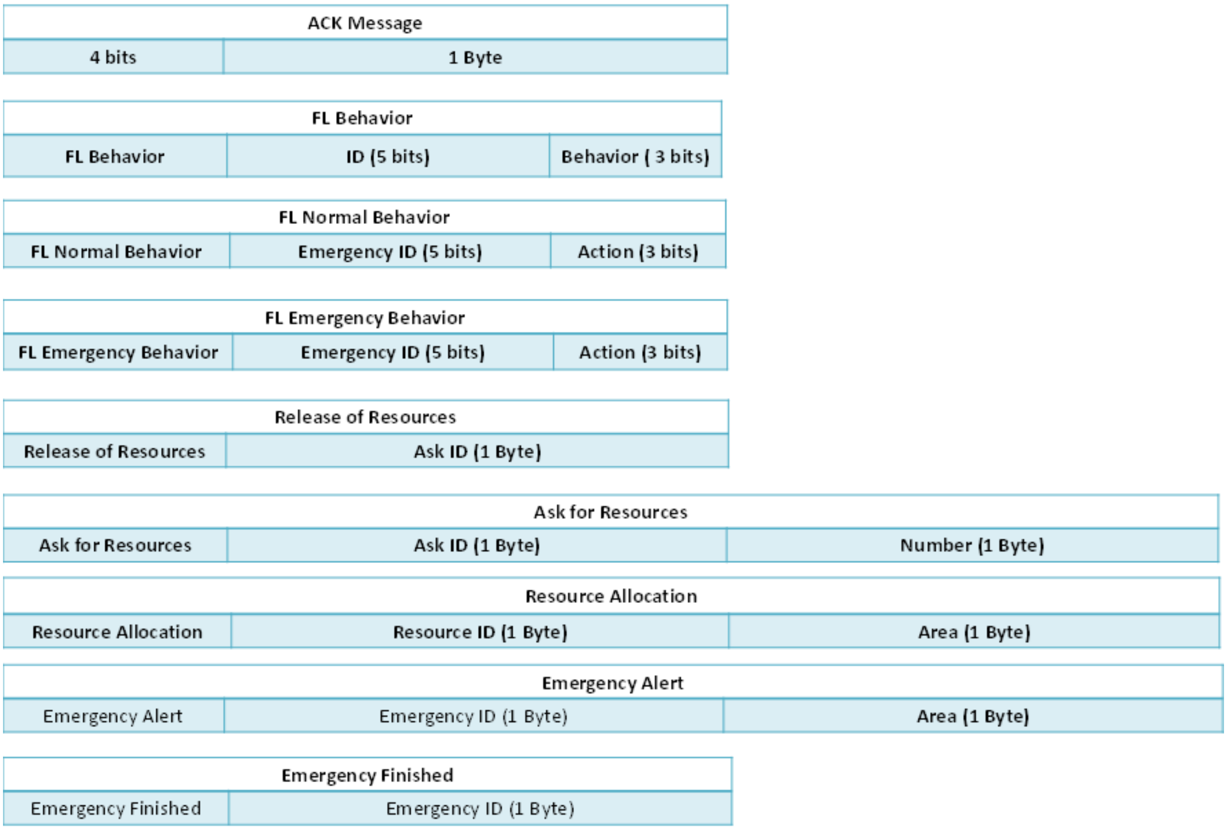
- FL Behavior: Assigns a concrete behavior to a traffic control node.
- FL Normal Behavior: Informs the CH that it must configure its nodes back to the normal behavior.
- FL Emergency Behavior: Notifies the CH that an emergency has been alerted and the controller will send FL Behavior messages to modify the workings of the nodes.

- Release of Resources: It is utilized to advise the local security service that the resources allocated to an emergency are not used anymore.
- Ask for Resources: Requests the allocation of local security resources to an emergency.
- Resource Allocation: Informs that the resource is assigned to the emergency. As well as the area ID and the number of resources that the service is going to allocate then.
- Emergency Alert: Message used by the emergency system to inform that there is an emergency.
- Emergency Finished: The emergency system uses this message to notify that the emergency is finished and the resources allocated to it have been released.

The value of the field Type indicates which of the previous messages is being sent. Then, the length field specifies the size of the Value field, where the important data is located. The ACK message has no value field and its length field it is equal to zero.

In order to invoke the local security unit service, the emergency has to be classified into high risk and low risk. This classification, which is done by the emergency service, it is notified by using the Emergency ID in the Emergency Alert message. The first bit of the field Emergency ID is set to 1 if the emergency is classified as high risk emergency. Therefore, there are 256 emergency IDs, 128 low risk IDs and 128 high risk emergency IDs.

The emergency ID is used to identify a concrete emergency during an established period of time. Both the controller and the emergency service can expand the emergency ID or generate a new one in their respective databases in order to manage them and enhance their applications or provide some new.



ACK Message	
4 bits	1 Byte

FL Behavior		
FL Behavior	ID (5 bits)	Behavior (3 bits)

FL Normal Behavior		
FL Normal Behavior	Emergency ID (5 bits)	Action (3 bits)

FL Emergency Behavior		
FL Emergency Behavior	Emergency ID (5 bits)	Action (3 bits)

Release of Resources	
Release of Resources	Ask ID (1 Byte)

Ask for Resources		
Ask for Resources	Ask ID (1 Byte)	Number (1 Byte)

Resource Allocation		
Resource Allocation	Resource ID (1 Byte)	Area (1 Byte)

Emergency Alert		
Emergency Alert	Emergency ID (1 Byte)	Area (1 Byte)

Emergency Finished	
Emergency Finished	Emergency ID (1 Byte)

Fig. 4 Message Structure

4. Energy Consumption Model

In this section, the energy consumption model used to measure the energy consumed in the entire emergency treatment process is described.

As is detailed on [26], the energy consumed per bit in the transmitter-receiver model can be modeled by the equation 1 to 5.

$$E_{bit} = E_{bit-tx} + E_{bit-out} + E_{bit-rx} + E_{bit-cod} \quad (1)$$

Where

$$E_{bit-tx} = \frac{P_{tx\_e}\left(\frac{header + payload + trailer}{rate}\right) + P_{tx\_ini}T_{tx\_ini}}{payload} \quad (2)$$

$$E_{bit-out} = \frac{P_{tx\_out}\left(\frac{header + payload + trailer}{rate}\right)}{payload} d^2 \quad (3)$$

$$E_{bit-rx} = \frac{P_{rx\_e}\left(\frac{header + payload + trailer}{rate}\right) + P_{rx\_ini}T_{rx\_ini}}{payload} \quad (4)$$

$$E_{bit-cod} = \frac{E_{coding} + E_{decoding}}{payload} \quad (5)$$

and

E_{bit-tx} = energy consumed per bit in electronic transmission

$E_{bit-out}$ = energy consumed per bit in output transmission

E_{bit-rx} = energy consumed per bit in electronic reception

$E_{bit-cod}$ = energy consumed per bit in coding/decoding processes

It is assumed that the energy dissipated in transmission (Ptx_e) is 50 nJ/bit, is exactly the same as the energy dissipated in electronic reception ($P\text{r}x\_e$). On the other hand, the energy that is consumed in output transmission is 100 pJ/bit/m2 [27].

In addition, the maximum energy consumed in coding and decoding, by using AES encryption, is 0.55 nJ/bit [28].

Attending to Figures 2 and 3, a node is involved in the following messages exchanged during an emergency situation in which the node belongs to an affected route:

- It receives a FL Behavior in order to modify the actions that it executes in its usual way of working from the CH.
- It also receives a FL Behavior from the CH for each neighbor that exists in the new route. This is done for malfunction utilities.
- It sends an ACK message to each neighbor. It also receives an ACK from every neighbor.

In order to reduce the energy consumption, the system uses UDP instead of TCP. This decision not only avoids the connection establishment mechanism or congestion control as it is not necessary for this proposal, but also reduces the bits transmitted. With the combination of IEEE 802.15.4 for communicating the nodes with the CH, IPv6 and UDP the size of the messages is between 80 and 82 bytes.

With this way of communicating we can define the energy consumed in communication as it is expressed in equation 6:

$$E_{com} = N_{Neig} * S_{Ack} * \left(E_{bit-tx} + E_{bit-out} + E_{coding}\right) + N_{Neig} * S_{Ack} * \left(E_{bit-rx} + E_{decoding}\right) + (N_{Neig} + 1) * S_{Beh} * \left(E_{bit-rx} + E_{decoding}\right) \quad (6)$$

Where

N_{Neig} = Number of Neighbors

S_{Ack} = Size of ACK message

S_{Beh} = Size of FL Behavior message

The first addend corresponds to the energy wasted in forwarding the ACK to every neighbor. The second one is related to the received ACKs from the neighbors. Finally, the last addend belongs to the energy consumed in the reception of the behavior messages from the CH.

5. Methodology and Results

In this section, the experiments performed to test our proposal are detailed. They are all executed in a scenario where a smart city is deployed exactly as the one showed in Fig 5. This smart city is treated as a SDN. The equivalent SDN is showed in Fig 6. In this figure, can be seen as the IoT networks are replaced by OpenFlow switches and the different neighbors are converted into PC hosts. The links between the Switches or PCs are the different roads and highways in the city. Fig. 6 shows also the different labels and IP addresses given to every node in the SDN. Nodes s1-s6 belong to the city itself and have some hosts connected to them. Switches s3 and s4 represent the inner-city. Furthermore, s7-s14 represents the beltway of the city. The color of the links describes the different scenarios where that roads are involved in the emergency. This is further explained in the subsection 5.2.

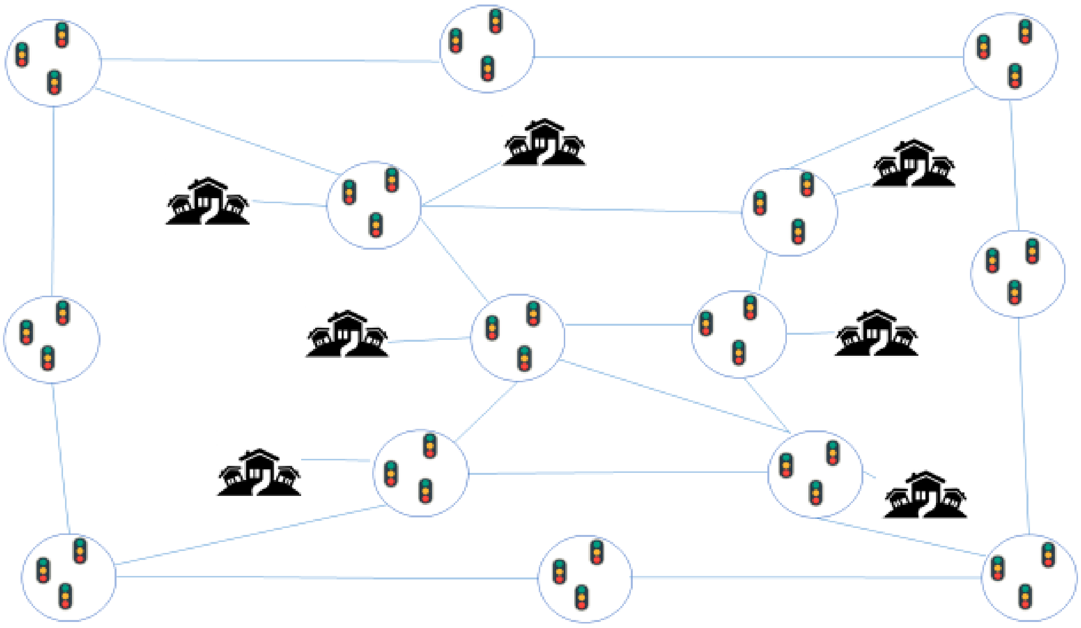


Fig. 5 Scheme of the city used for testing

The experiments are run by using the emulator Mininet, which provides an easy way to test the SDN by emulating the Switches as Linux Hosts using OpenVSwitch. The urban traffic (emergency and regular traffic) corresponds to packets in the SDN.

The experiment is divided into two parts. The first part compares the behavior of real urban traffic during an emergency in both situations, with the performance of our proposal and without it. The second part measures the energy consumed in three different scenarios depending on the number of nodes and the neighbors in the IoT networks. Each experiment is described in the following subsections.

5.1 Traffic comparison

The traffic in the city is simulated as packets that go through the network. In this first test several multimedia streams are launched in order to emulate normal traffic in a city. The delay is measured as the time the cars need to get to their destination. Then, an emergency occurs and some emergency traffic is sent to the emergency area. The delay of the emergency resources are also measured.

This experiment is run in two different scenarios. First, the emergency happens and no algorithm is functioning, so our proposal does not provide any solution. This is the emulation of the usual way of working with emergencies.

On the other hand, in the second scenario, we measured also the delay that both the emergency and the normal traffic have when the proposal provides solutions for emergency priority routes and normal traffic diversion.

In order to achieve this, an emergency in the area emulated as the switch s3 is launched. This emergency is emulated as a congestion in the links in red as seen in Fig. 6. The links in other colors are not congested. The meaning of these colors is explained in the following experiment and they are not relevant in the first one.

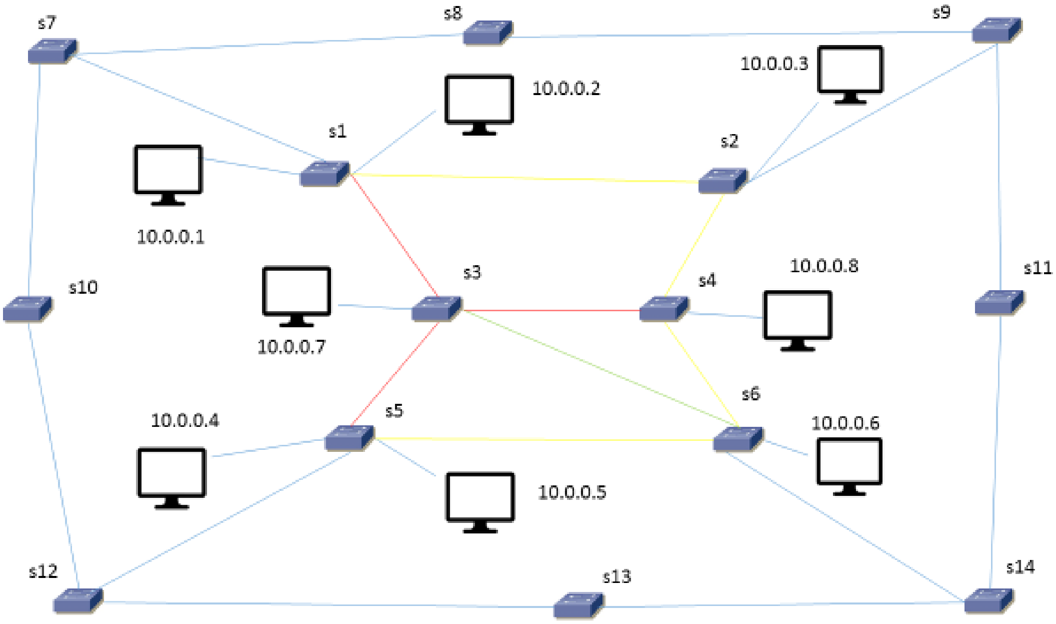


Fig. 6 SDN corresponding to the Smart City

When the emergency is launched, the emergency traffic is sent. In the first scenario, it shares the congested links with the normal traffic along the entire path to s3. However, when the algorithm is running, it forwards the normal traffic through s2.

The dynamic modification of the path used to forward the normal traffic that the algorithm uses, together with the priority of the emergency results, implies a variation of the delay of both types of traffic. This is showed in Fig. 7 and Fig. 8, where the delay of normal traffic is displayed, and in Fig. 9 and Fig. 10, where the delay of the emergency traffic is presented. In the normal traffic graphs, the x-axis represents the packet number in the multimedia stream. However, in the emergency traffic, the resources are treated individually and the x-axis represents the moment in time after the emergency occurs, when the resource is sent. In Fig. 7 and Fig. 9 the traffic is sent in the first scenario, without algorithm and, in Fig. 8 and Fig. 10, the algorithm is working.

In Fig. 7, the normal traffic is being sent in the first scenario in normal conditions until the packet number 1366 approximately, where the emergency happens. In Fig. 8 the normal traffic is suffering the congestion and then, the algorithm modifies the route paths. Both graphs show the change that delay values experience.

In the first scenario, regarding the normal traffic, it starts with normal values of delay of 20 ms and sometimes around 100 (due to the variation of the multimedia bitrate). The average delay, however, is around 10 ms. When the emergency occurs, the links are slightly congested and shared with emergency traffic. The average delay increments to more than 130 ms.

In the second scenario, the normal traffic experiments a delay just like the delay in scenario 1, but when the algorithm changes the path towards one slower and longer but not congested route, the delay decreases, achieving an average delay of 67 ms. The average delay is greater than the delay obtained when there is no emergency in the path, but is significant lower than the one obtained when there is no algorithm, a 50% of average delay reduction approximately.

Regarding the emergency traffic, Fig. 9, related to the first scenario, shows that the average delay is 26ms. When the algorithm is running, in Fig. 10, the average delay decreases to 17ms. There is a 33% of average delay reduction.

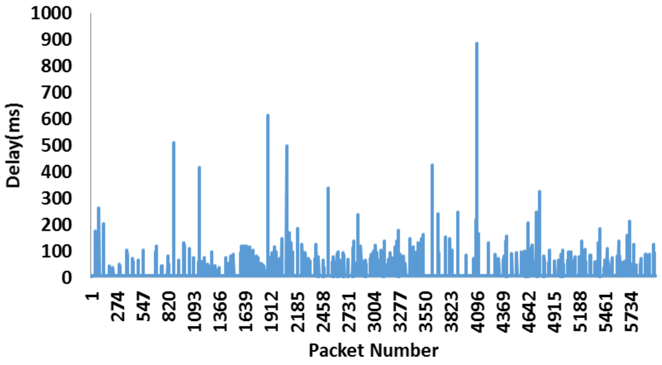


Fig. 7 Delay in normal traffic sending without system application.

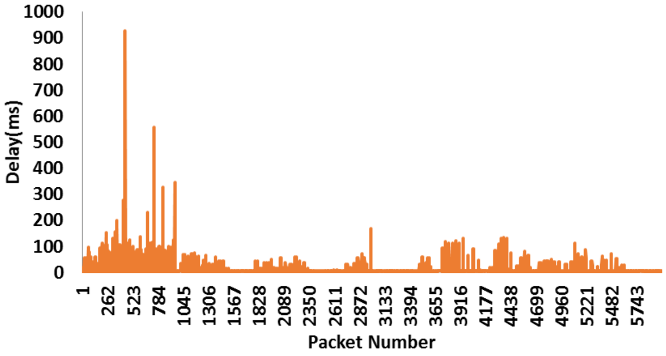


Fig. 8 Delay in normal traffic sending with system application

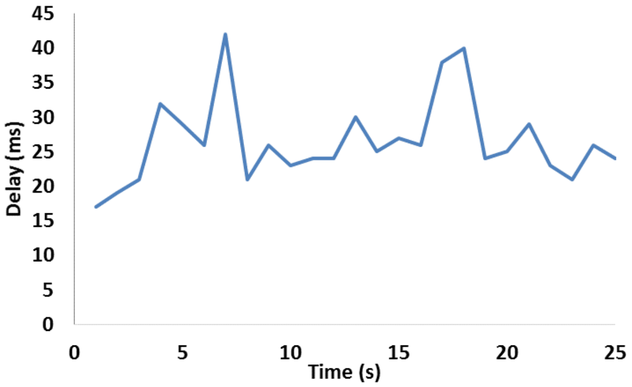


Fig. 9 Delay in emergency traffic without system application

5.2 Energy consumption

The second experiment, whose aim is to measure the energy consumed in the communication of the nodes, is executed in three different scenarios. These scenarios differ in the size of the emergency in terms of links congested in the network. The links that are affected in each emergency are colored differently as seen in Fig. 6. The red ones are affected in the three scenarios, even the third one which is the one used in the previous subsection, where only the red links are affected. Scenario 2 includes the green link. çit increments the emergency damage isolating the neighbor connected to s3. Finally, by including the yellow links we obtain Scenario 1, where there is a high risk emergency and the inner-city paths are closed. The traffic is forwarded through the beltway.

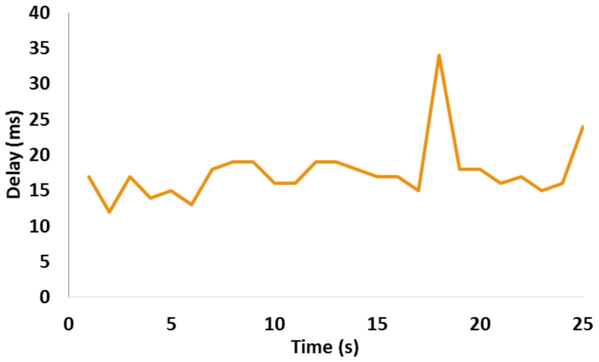


Fig. 10 Delay in emergency traffic with system application

These three scenarios involve a different number of communications between the nodes. The energy consumed per node and the total energy consumed in the nodes due to the emergency has been measured in terms of mJ attending to the equations presented in Section 4. Moreover, the number of neighbors has been increased in each individual simulation and the total has been calculated depending on the size of the IoT networks.

In Fig.11 the individual energy consumption per node is showed. The difference between the two last scenarios regarding the individual energy consumed is only significant when the number of neighbors increases to 5. The bigger difference appears in Scenario 1, with much more nodes involved. The increment of energy consumption is linear with the number of neighbors, which implies that the proposal is scalable. The energy in Scenario 3 increments from 37mJ with 1 single neighbor to 185mj with 6 Neighbors.

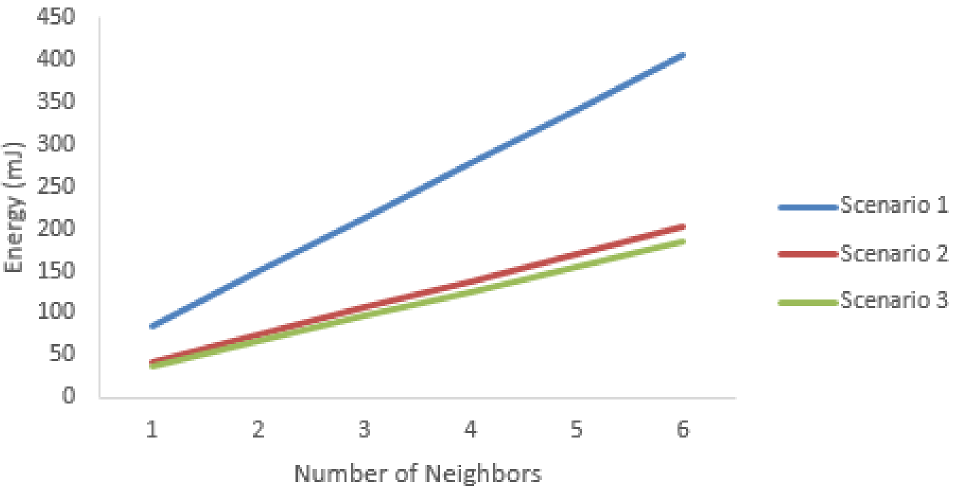


Fig. 11 Energy consumed per node

Figs. 12 to 14 are related to total energy consumed. The energy in every figure is calculated with a different number of neighbors. The x-axis in these graphs are the number of nodes in every IoT network.

The increment of the number on nodes increases linearly the total energy consumed in the exchange of messages. The three scenarios described are painted in different colors. The difference between scenario 2 and scenario 3 is not as significant as the difference between scenario 2 and scenario 1. The energy consumed in scenario 1 (6698mJ) with 3 nodes is more than 3 times the energy wasted in scenario 2 (2009mJ). When 10 nodes in the IoT network the difference increases, being 22327mJ the energy in Scenario 1 and 6326mJ in Scenario 2.

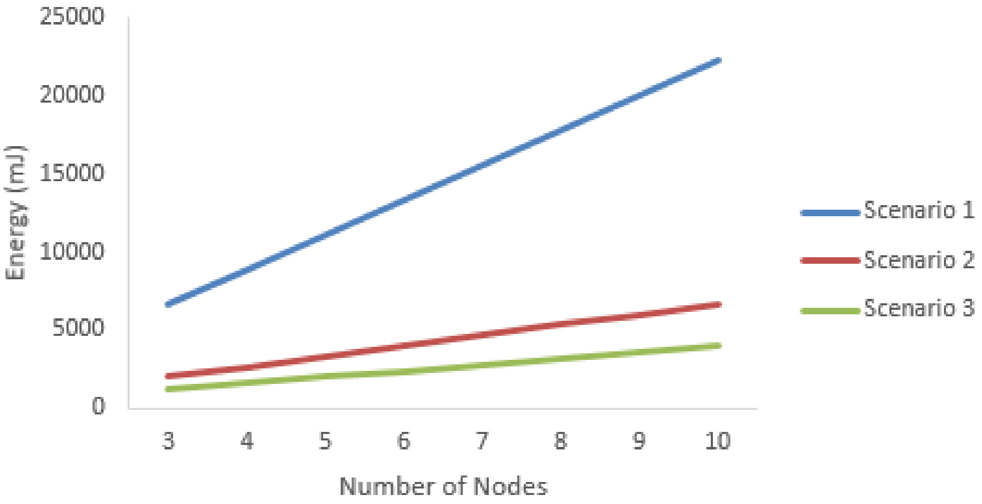


Fig. 12 Total energy consumed per scenario with 2 neighbor

As Fig.13 shows, with 4 neighbor in each IoT network the energy consumed increases, but the results are similar to the presentd in Fig. 12. Scenario 3 consumes 7566mJ with 10 nodes and scenario 2 12487mJ. Finally, scenario 1 wastes 41622mJ, almost 4 times the energy needed in scenario 2. Moreover, the energy consumed has increased near a 100% from the consumed with 2 neighbors. From example, in Scenario 2, with 2 neighbors the energy wasted is between 3349mJ with five nodes and 6698mJ with 10 nodes in every IoT network. However, with 4 neighbors the energy wasted is between 6243mJ and 12487mJ.

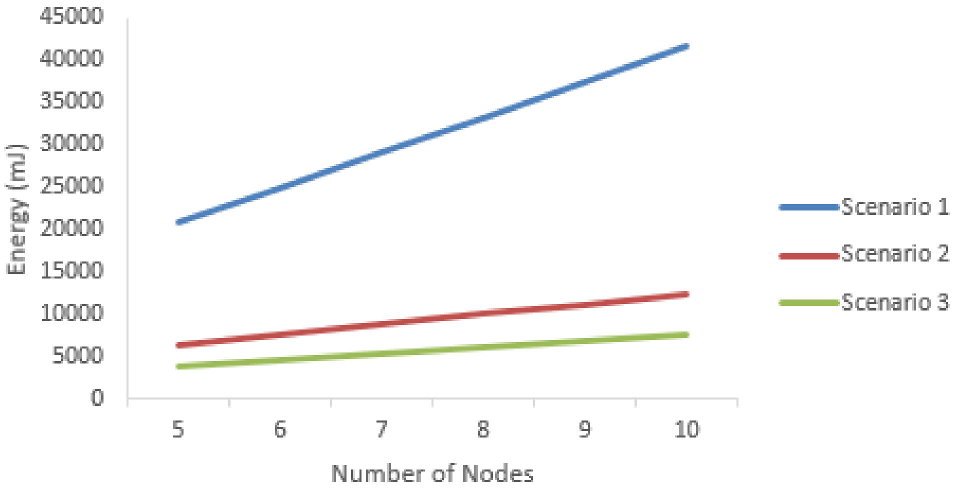


Fig. 13 Total energy consumed per scenario with 4 neighbors

Finally, Fig. 14 contains the results obtained with 6 neighbors.The results show how the difference between scenario 1 and the rest is greater than before, passing the 60000mJ in scenario 1 and scenarios 2 and 3 consuming between 10000mJ and 20000mJ.

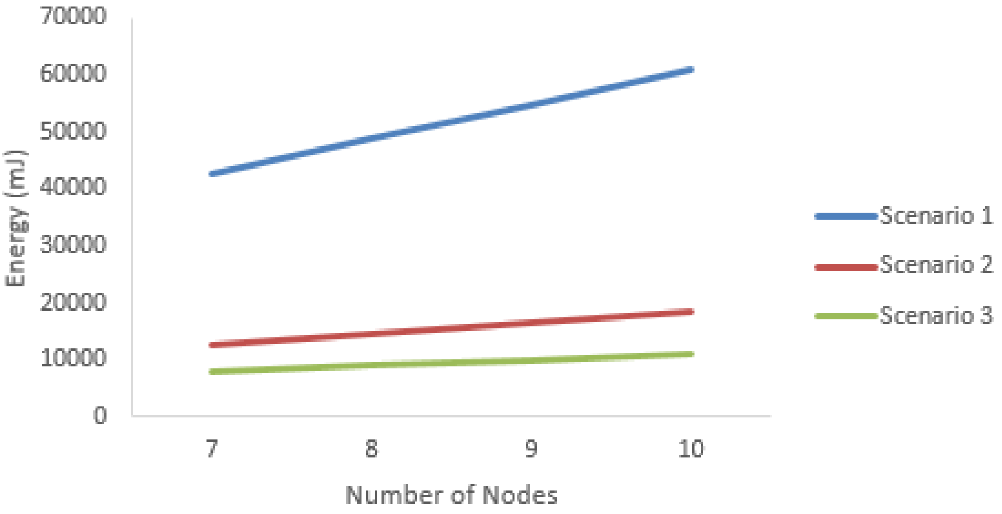


Fig. 14 Total energy consumed per scenario with 6 neighbors

6. Conclusion and Future work

Smart cities provide new applications based on IoT technology. Moreover, SDNs offer the possibility to control the network based on application requirements. In this paper, we combine this two technologies in order to take advantage of this possibility and propose a new application to manage emergency resources. This application uses the adaptability of the SDN in order to forward urban traffic using intelligent traffic lights and other traffic elements that are connected to IoT networks. The SDN controller communicates with cloud and local services so as to gather the data needed to make the best changes in order to manage the emergency.

Experiments show that the average time that emergency resources lose in getting to the emergency area is reduced in 33%, from 26ms to 17ms. This is due to the priority applied and path sharing avoidance. In terms of normal traffic, there is an increment of time when this traffic is forwarded toward a longer alternative path, but this time is lower than the time it takes to get to the destination when the emergency area is becoming congested. The average time difference is 50% between the alternative path and the congested path.

Furthermore, the communication protocol in case of emergency has been tested in terms of energy consumption. Based on the energy model described in Section 4, experiments illustrate that the energy consumed increases linearly with the number of nodes and neighbors in the IoT networks, which indicates that the proposal is scalable and could be applied in every kind of smart city.

As future work, the proposal can be either enhanced or tested in other kind of Smart Cities. Emergency detection has not been discussed in the paper. The use of cameras or other detection system could be added to the IoT networks and a detection system could be added to the SDN network. This system could communicate with the cloud emergency system to inform about emergency or data treatment. Artificial intelligence combined with our proposal would be able to provide a mechanism to learn what type of traffic path changes provide the best solution in terms of the time consumed to get the destination area. Finally, we would like to integrate some sensors to monitor air pollution [16] in order to use our system in polluted cities and modify the traffic [29]. Finally, we will analyze the possibility of combining the proposed architecture with 5G network [30].

Acknowledges

This work has been partially supported by the “Ministerio de Educación, Cultura y Deporte”, through the “Ayudas para contratos predoctorales de Formación del Profesorado Universitario FPU (Convocatoria 2015)”. Grant number FPU15/06837, by the “Ministerio de Economía y Competitividad”, through the “Convocatoria 2014. Proyectos I+D - Programa Estatal de Investigación Científica y Técnica de Excelencia” in the “Subprograma Estatal de Generación de Conocimiento”, project TIN2014-57991-C3-

1-P and through the "Convocatoria 2016 - Proyectos I+D+I - Programa Estatal De Investigación, Desarrollo e Innovación Orientada a los retos de la sociedad" (Project TEC2016-76795-C6-4-R).

References

[1] T. Nam and T. A. Pardo, "Conceptualizing Smart City with Dimensions of Technology, People and Institutions", Proceedings of the 12th Annual International Conference on Digital Government Research, College Park, Maryland, USA, 12-15 June 2011, pp. 282-291.

[2] L. Hernandez, C. Baladron, J.M. Aguiar, B. Carro, A. Sanchez-Esguevillas, J. Lloret, D. Chinarro, J. J. Gomez-Sanz and D. Cook, "A multi-agent system architecture for smart grid management and forecasting of energy demand in virtual power plants", IEEE Communications Society, Vol. 51, No. 1, pp. 106-113, January 2013.

[3] J. Steenbruggen, P. Nijkamp and M. Van der Vlist, "Urban traffic incident management in a digital society: An actor-network approach in information technology use in urban Europe", Technological Forecasting and Social Change, Vol. 89, pp. 245-261, November 2014.

[4] A. Cuesta, O. Abreu and D. Alvear, "Future Challenges in Evacuation Modelling", Evacuation Modelling Trends, Springer, Cham, 2014. DOI: https://doi.org/10.1007/978-3-319-20708-7_5

[5] H. El-Sayed, S. Sankar, M. Prasad, D. Puthal, A. Gupta, M. Mohanty and C. Lin, "Edge of Things: The Big Picture on the Integration of Edge, IoT and the Cloud in a Distributed Computing Environment", IEEE Access, published online, pp. 1-14, 2017. DOI: https://doi.org/10.1109/ACCESS.2017.2780087

[6] K. A. C. Basconcillo, D. J. B. Benitez, E. A. S. Cantuba, R. E. L. Enríquez, C. R. I. Falcon, K. K. D. Serrano, E. C. Guevara and R. R. P. Vicerra, "Development of a vehicle and pedestrian simulation environment with M.I.S.O fuzzy logic controlled intelligent traffic light system", 5th International Conference on Information and Communication Technology, Malacca City, Malaysia, 17-19 May 2017, pp. 1-6.

[7] S. Sharma, A. Pithora, G. Gupta, M. Goel and M. Sinha, "Traffic Light Priority Control For Emergency Vehicle Using RFID", International Journal of Innovations in Engineering and Technology, Vol. 2, No. 2, pp. 363-366, April 2013.

[8] K. Z. Ghafoor, K. A. Bakar, J. Lloret, R. H. Khokhar and K. C. Lee, "Intelligent beaconless geographical forwarding for urban vehicular environments", Wireless Networks, Vl. 19, No. 3, pp. 345-362, April 2013.

[9] 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, August 2015.

[10] S. Sendra, A. Rego, J. Lloret, J. M. Jimenez and O. Romero, "Including artificial intelligence in a routing protocol using Software Defined Networks", IEEE International Conference on Communications Workshops, Paris, France, 21-25 May 2017, pp. 670-674.

[11] M. Taha, L. García, J. M. Jimenez and J. Lloret, "SDN-based throughput allocation in wireless networks for heterogeneous adaptive video streaming applications", 13th International Wireless Communications and Mobile Computing Conference, Valencia, Spain, 26-30 June 2017, pp. 963-968.

[12] S. Tomovic, M. Pejanovic-Djurisic and I. Radusinovic, "SDN Based Mobile Networks: Concepts and Benefits", Wireless Personal Communications, Vol. 78, No. 3, pp. 1629-1644, October 2014.

[13] E. Ahmed, I. Yaqoob, A. Gani, M. Imran and Mohsen Guizani, "Internet-of-Things-Based Smart Environments: State of the ART, Taxonomy, and Open Research Challenges", IEEE Wireless Communications, Vol. 23, No. 5, pp. 10-16, November 2016.

[14] K. Nellore and G. P. Hancke, "A Survey on Urban Traffic Management System Using Wireless Sensor Networks", Sensors, Vol. 16, No. 2 p. 157, January 2016.

[15] C. Diakaki, M. Papageorgiou, I. Papamichail and I. Nikolos, “Overview and analysis of Vehicle Automation and Communication Systems from a motorway traffic management perspective”, Transportation Research Part A: Policy and Practice, Vol. 75, pp. 147-165, 2015.

[16] M. Temani, R. Agarwal, R. Kholi and M. Sen, “Smart Approach to Traffic Management using LabView”, Fifth international Conference on Intelligent Systems, Modelling and Simulation, Langkawi, Malaysia, 27-29 January 2014, pp. 68-71.

[17] X. Zhou, S. Tanvir, H. Lei, J. Taylor, B. Liu, N. M. Rouphail and H. C. Frey, “Integrating a simplified emission estimation model and mesoscopic dynamic simulator to efficiently evaluate emission impacts of traffic management strategies”, Transportation Research Part D: Transport and environment, Vol. 37, pp. 123-136, 2015.

[18] Md. A. Samad Kamal, J. Imura, T. Hayakawa, A. Ohata and K. Aihara, “Smart Driving of a Vehicle Using Model Predictive Control for Improving Traffic Flow”, IEEE Transactions on Intelligent Transportation Systems, Vol. 15, No. 2, April 2014, pp. 878-888.

[19] L. Calderoni, D. Maio and S. Rovis, “Deploying a network of smart cameras for traffic monitoring on a “city kernel””, Expert Systems with Applications, Vol. 41, No. 2, pp. 502-507, 2014.

[20] A. Kanungo, A. Sharma and C. Singla, “Smart Traffic Lights Switching and Traffic Density Calculation using Video Processing”, Recent Advances in Engineering and Computational Sciences, Chandigarh, India, 6-8 March 2014, pp. 1-6.

[21] Z. Cao, S. Jiang, J. Zhang and H. Guo, “A Unified Framework for Vehicle Rerouting and Traffic Light Control to Reduce Traffic Congestion”, IEEE Transactions on Intelligent Transportation Systems, Vol. 18, No. 7, pp.1958-1973, July 2017.

[22] H. M. Kammoun, I. Kallel, J. Casillas, A. Abraham and A. M. Alimi, “Adapt-Traf: An adaptive multiagent road traffic management system based on hybrid ant-hierarchical fuzzy model”, Transportation: Research Part C: Emerging Technologies, Vol. 42, pp. 147-167, 2014.

[23] A. Artikis, M. Weidlich, F. Schnitzler, I. Boutsis, T. Liebig, N. Piatkowski, C. Bockermann, K. Morik, V. Kalogeraki, J. Marecek, A. Gal, S. MAnnor, D. Kinane and D. Gunopoulos, “Heterogeneous Stream Processing and Crowdsourcing for Urban Traffic Management”, Proceedings of the 17th International Conference on Extending Database Technology, Athens, Greece, 2014, pp. 712-723.

[24] J. Joshi, K. Jain, Y. Agarwal, M. J. Deka and P. Tuteja, “TMaaS: Traffic Management as a Service Using Cloud in VANETs”, IEEE 3rd International Conference on Smart Instrumentation, Measurement and Applications, Putrajaya, Malaysia, 24-25 November 2015, pp. 1-6.

[25] R. Bauza and J. Gozalvez, “Traffic Congestion detection in large-scale scenarios using vehicle-to-vehicle communications”, Journal of Network and Computer Applications, Vol. 36, No. 5, pp. 1295-1307, 2013.

[26] S. Sandra, J. Lloret, M. Garcia, J.F. Toledo, “Power saving and energy optimization techniques for Wireless Sensor Networks”, Journal of Communications, Vol. 6, No. 6, 2011.

[27] W. R. Heinzelman, A. Chandrakasan, A., H. Balakrishnan, “Energy-Efficient Communication Protocol for Wireless Microsensor Networks”, In Proceedings of the IEEE 33rd annual Hawaii international conference on System Sciences, Maui, Hawaii, January 4-7, 2000.

[28] J.P. Kaps, B. Sunar, “Energy Comparison of AES and SHA-1 for Ubiquitous Computing”, In Proceedings of the EUC 2006 Workshops: NCUS, SecUbiq, USN, TRUST, ESO, and MSA, Seoul, South Korea, August 1-4, 2006.

[29] H. N. AL-Hashimi and, W. N. Hussein, “PMIPv6 Assistive Cross-Layer Design to reduce handover latency in VANET Mobility for Next Generation Wireless Networks”, Network Protocols and Algorithms, 2015, Vol 7, No 3, pp.1-17.

[30] V. Perez-Garrancho, L. Garcia, S. Sendra, J. Lloret, Mobile Low Cost system for Environmental Monitoring in Emergency Situations, In proc. of the the Seventh International Conference on Ambient Computing, Applications, Services and Technologies (AMBIENT 2017), November 12 - 16, 2017 - Barcelona, Spain.