

Towards a Distributed SMA-based Solution for the Interoperability of Hospital Information Systems for Better Routing of Emergency Ambulances

<https://doi.org/10.3991/ijoe.v17i12.25455>

Meryem Fakhouri Amr^(✉), Mouhcine Elgarej, Nezha Benmoussa,
Khalifa Mansouri, Mohammed Qbadou
University Hassan II, Casablanca, Morocco
meryemfakhr@gmail.com

Abstract—Multi-agent systems MASs have been widely used to interoperate hospital information systems (HISs). The use of MASs for HISs interoperability has become a central solution, especially in the field of emergency medicine. In emergencies, the notion of delay is relative, because responders only have a few minutes to react. This emergency response time has an important role in the event that an accident occurs on the road. Existing procedures for the emergency ambulance (EA) dispatch strategy are based on manual dispatch. In this work, we are introducing a distributed emergency ambulance (DEA) routing system to control emergency latency time, which includes driving route planning to guide emergency vehicles and the allocation of distributed emergency resources (emergency ambulances and hospitals) to reduce the EA response time caused by traffic or the wrong human decision to transport ambulance to the accident site. The allocation of resources (hospitals) is ensured through a recommendation system based on the interoperability of several interconnected HISs using a multi-agent system. The proposed solution takes into consideration dynamic traffic flow information during the day to build dynamic paths for EA. The improved method is based on a distributed architecture to calculate and find the optimal pathway for a set of emergency vehicles based on ACO ant colony optimization techniques. The results of the simulation show that the proposed method can decrease the total travel time of the ambulance to reach the accident position compared to conventional methods that use lights and sirens to warn other vehicles to free up the road for the ambulance or use a traditional approach based on the vision/reflection of the driver to choose in a random way the paths to take. Based on such a solution, ambulance staff will be able to save lives by optimizing the total journey with the minimum travel.

Keywords—multi-agent systems (MASs), interoperability, ant colony optimization (ACO), hospital information systems (HISs), distributed emergency ambulance (DEA)

1 Introduction

Many areas suffer from the catastrophic state of their infrastructure. The access to the accident site is considered a complex mission [1]. [2] In addition to this, the other problem is to find the closest hospital to the accident site that can accommodate the person in emergency. The main problem is, therefore, to manage the available ambulance and to prepare the optimal path that will be followed to arrive at the accident site and then return to the nearest hospital in order to rescue the injured person or in need of emergency intervention [3]. When the ambulance receives information about the accident, it travels from its current position to a final destination that corresponds to the accident location. Moreover, the nearest hospital unit will assist these wounded depending on the severity of their condition. Most ambulances use lights and sirens to warn drivers of their presence. However, these drivers cannot react effectively or react too late incorrectly because the sound of sirens is not clearly captured. Most of the time, even if the drivers see the ambulance, they will find it difficult to get away from the emergency vehicle in the event of a traffic jam or rush hour. In the literature, many works [4], [5], [6] has focused on selecting the fastest trajectory that corresponds to the shortest distance between a set of points. In the routing of ambulances, we need to find the optimal route that takes into consideration a set of constraints for instance speed limits and traffic jams and other factors... Current emergency ambulance routing systems are based solely on human assessments, the most existing strategy such as planning the route from ambulance/accident station and from accident/hospital, the selection of hospital units is based on traditional distributions which increase latency to meet the emergency request. The main idea of this article is to find the fastest path by proposing a new distributed system based on the ACO algorithm (Ant Colony Optimization) [7], [8], [9] to get the shortest route according to a set of intelligent agents dedicated to helping the ambulance to arrive at the accident site and pick up the injured person at the nearest hospital. The ant colony system algorithm is a distributed algorithm [10], [11], self-adapted based on the behavior of real ants. It is based on a probability transition and a simple mathematical model that is easy to implement in several systems. ACO algorithms have achieved satisfactory feedback both in terms of temporal complexity and search efficiency compared to heuristic methods [12]. This algorithm is considered to be a distributed system in which ants work in parallel to provide positive feedback and has been applied to find a solution to problems related to transportation [13] and combinatorial optimization and communication networks [14], [15], [16].

Our solution will be based on a distributed environment in which several agents will work together in parallel to guide the ambulance and find all the resources thanks to the interoperability of hospital information systems with a central control and supervision system whose objective is to save the person's life. The proposed system will allow us to generate dynamic route planning for the ambulance [17], [18] based on a set of factors for instance speed limits and traffic jams, Road conditions to calculate alternative routes and guide the vehicle. The emergency routing system proposed in this work includes the following main elements: distribution of emergency vehicle resources, recommendation system to find the nearest appropriate hospital according to the condition of the injured person, route planning for an emergency ambulance. The results of the

simulation show that the proposed solution contributes in an effective way to reducing the emergency time by arriving on time at the accident site and that the emergency crew are able to save human lives in the fastest possible way. The rest of this work is organized as follows: a literature review on related work followed by the description of the emergency ambulance life cycle. Then the overview of the emergency vehicle routing problem will be presented in the next section. The concept of ACO and the proposed distributed strategy for the dispatch of emergency ambulances will be described in detail for further clarification. Finally, we will present the simulation results as well as the corresponding conclusion.

2 Related works

The key function of our system is to facilitate the timely sharing of information. Different actors from different systems must communicate, collaborate and access information with ease [19]. For this, interoperability between the components of our architecture is crucial and it can be guaranteed by multi-agent-based [20] systems while respecting their autonomy. Lack of interoperability can create completely isolated systems. These systems must exchange data among themselves. Several research studies have addressed the usefulness of multi-agent systems for the interoperability of hospital information systems. [21] proposed an agent-oriented architecture to address the problem of data heterogeneity between health information systems. The proposed model is based on a typical emergency response scenario where a first medical assistance team urgently needs to retrieve a citizen's Patient Index using mobile devices, which is part of their electronic health record.

Also [22] proposed an agent-based platform called "Biomedical Multi-agent Platform for Interoperability" to solve the problem of interoperability between health systems. The main objective of this platform is to facilitate communication between the agents of a multi-agent system. It also facilitates interaction between humans and agents through an interface that allows administrators to add new agents and monitor their activities in real time. The use of multi-agent systems in healthcare has been presented as a complementary technique to improve the performance of systems in terms of interoperability, scalability and reconfiguration. We intend to use the agent paradigm for the implementation of the emergency ambulance routing system while ensuring interoperability between all the players in the system.

The work of [23], aims to provide an effective way to control emergency ambulance routing requests to save the lives of patients in the event of a disaster taking into consideration the traffic route and the transportation route of the injured from the accident site to the hospital. The main work focused on minimizing the response time of emergency medical personnel, the problem was modeled as an open vehicle route problem but with some specifications like picking up casualties from multiple locations and delivering them to specific hospitals.

The strategy presented is based on two levels: the aggregation of data to build the research environment (by collecting information on ambulance and patients) and on the second level, they focus on optimizing the route followed by the ambulance using the Petal algorithm and swarm optimization techniques to find the optimal paths for

emergency ambulances. In the real world, [24] Emergency Medical Services (EMS) vehicles (ambulances) are created to provide emergency medical assistance to the patient at various locations or to pick them up at the nearest hospital. This ambulance is equipped with several tools that can be used to provide emergency care to the injured person. In this manuscript [25], the authors focused on the management of the available ambulance that can intervene in these emergency cases besides on the control of the time necessary to arrive at the accident site. To do this, they are based on the A* algorithm to calculate the planning of the route that will be followed by the ambulance to arrive as soon as possible and they use the road network to draw the map that will be used to travel without taking consider the traffic route on this map. In the article [26], the authors propose a new centralized framework for intelligent navigation of emergency vehicles in the event of a disaster to guide vehicles according to the dynamic density of traffic flow. In this procedure, an improved Fuzzy Control Ant Colony System (FCACS) search strategy is presented to find the best route planning for emergency vehicles from the disaster site to the nearest hospitals. An intelligent traffic system is also integrated into this framework to accelerate the convergence of the solution, to help ambulances on the road by controlling the traffic light.

In addition, the authors of the article [27] propose a new route planning strategy for the emergency ambulance transport of rural patients requiring continuous remote monitoring. The model is mathematically considered an NP-difficult graph problem and aims to find an optimal solution to find the shortest path and maintain the online communication coverage to help us in the patient follow-up. They design their algorithms and conduct preliminary experiments to find optimal planning techniques that work well together to improve emergency patient transfer scenarios and ensure the best communication system to monitor the patient's condition during the journey.

3 Emergency ambulance life cycle

Our ambulance routing model is a dynamic system that aims to minimize the total time consumed during an emergency response [28]. The objective is to manage the total delay of emergency interventions from the moment T_c (Time of the call) in which we receive the emergency requests (call) to the time T_a (Time of arrival) which represent the time of arrival of the ambulance at the accident site [29], ending with the time taken to pick up the patient from the nearest hospital. The model we propose is based on two levels of urgency:

- **The first level:** we start at the tactical level, including ensuring the best allocation of ambulances available in hospitals.
- **The second level:** we end with an operational level that generates a trajectory plan for the ambulance during an emergency call.

Once a call arrives, it will be handled by the ShoreControl (SC), which will contact the Ambulance Monitoring AM to search for the nearest ambulance to the accident site. The AM is looking for the closest ambulances to the scene of the accident according to the GPS coordinates communicated by the SC. He then sends the list of the closest

ambulances found to the SC who will contact the routing center to calculate the shortest path of each of these ambulances in relation to the place of the accident, taking into account the factor of road traffic. The routing center RC, relying on data from Google Maps and the calculation results of the ACO algorithm based on the ant colony system return the list of the shortest paths between ambulances and the the accident site. Route planning will be generated at the given ambulance to help them arrive on time. This trajectory will be dynamically generated according to the traffic route, which means that at every moment a new path will be created for the ambulance. When the ambulance arrives at the accident site, ambulance staff will assess the patient's condition and send a preliminary diagnosis of the patient's condition to the ShoreControl SC. According to this diagnosis, the (SC) establishes a keyword list concerning the patient's condition and contacts the Hospital Center (HC) to request the list of nearest hospitals that are specialized in this area they can help the patient in emergency. The HC connects to the central database (CDB) to search for hospitals that are spreading the criteria sent by the SC. The HC chooses the hospitals available to accommodate the patient and returns the list to the HC. The SC contacts the routing center to calculate the shortest path from the current ambulance position to the hospitals by applying the ACO algorithm. Then, the SC receives the list of the nearest hospitals sent by the routing center and chooses a hospital which by the list according to these two criteria: distance/travel time. Finally, the best hospital is selected and sent to the ambulance and a new path will be generated to guide it from its current position to the selected hospital.

4 Emergency ambulance routing

In order to avoid emergency ambulance delivery problems, we propose a strategy based on three main elements that we describe below:

- **Allocation of emergency ambulance resources**
- **Allocation of the shortest available hospital**
- **Vehicle routing for EA**

In this architecture (Figure 1), we assume that there are multiple Road Sensor Units (RSU) installed at the roadside and at intersections. The RSU is used to count all vehicles crossed on this road. The synchronization between the different RSU on the road network will be useful to control the number of vehicles existing between two adjacent RSU. In our work, we take the number of stationary vehicles as the traffic flow on this route and this parameter will be used by our distributed system to find the best routes with less traffic jams. We know that the distance between two adjacent RSU is predefined and that the number of stationary vehicles is also fixed, so to define the maximum number of vehicles on this route, we can use this simple rule:

$$T_{density} = \frac{\text{The length of route between two adjacent RSU}}{\text{Surface taken by a vehicle}} \quad (1)$$

All this data is shared with a routing center which manages the traffic flow. Each EA is equipped with a GPS (Global Position System) and has its own OBD-II (Onboard

Diagnostics) system that manages and monitors the status of the vehicle (current speed, fuel quantity, etc...). The information collected on the condition of the road is used to calculate a dynamic trajectory plan for the ambulance at each time.

The emergency routing system is based on the actual information received from the sensors on road condition and traffic, the number of emergency vehicles that can collaborate to serve the citizens and the hospitals available with their specialty and areas emergency that can manage to serve people. All of these input parameters are used to calculate the best path between the current location of the selected ambulance and the accident position as shown in the following figure:

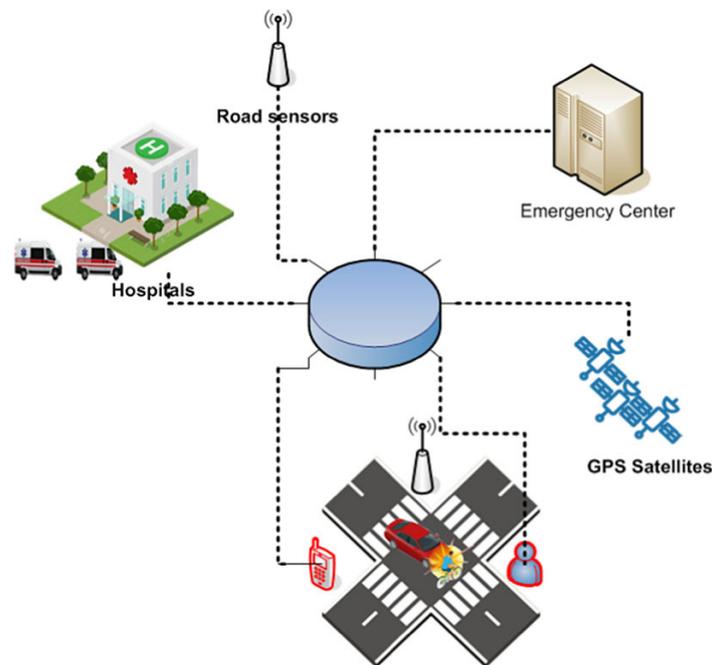


Fig. 1. Emergency ambulance routing architecture

4.1 Preparation of the collaborative environment and provision of emergency resources

Many researches have focused only on planning the path to follow for EA, but the main problem is to manage the available emergency resources according to the needs of the system. In real life, the distribution of emergency ambulances is based on manual procedures, we have human staff who control and dispatch ambulances to reported accidents. Sometimes the proposed paths are not optimal or they are static and do not take into account the traffic flow on the road, which causes more delay in arriving at the accident. On the other hand, we have to choose the nearest and most efficient hospital that can serve the patient in an emergency or we have to look for another hospital that can help them.

After receiving an emergency call, the ambulance monitoring team (AM) will save the GPS position of the accident and select the nearest ambulance. Then, the routing system generates a preliminary plan of the path to send the current vehicle to the accident based on the available information on the current road condition and traffic flow. The procedure for finding the shortest path to the accident area is shown in Figure 5. For each selected ambulance, the routing center will estimate the shortest travel time that will be consumed between the location of the ambulance and the position of the accident. The proposed procedure will estimate the travel time of each ambulance based on actual traffic conditions and previous trajectories to construct the optimum path to the accident site. The scenario for sending an ambulance can be summarized as follows:

Level 1: After an emergency call is handled, the ShoreControl will analyze the situation and set the number of vehicles to respond to this accident, the system will also find the closest ambulance to the accident area.

Level 2: Depending on the location of the ambulance and accident, the system will calculate the time required to travel to the accident site for each selected ambulance.

Level 3: When ambulance personnel analyze the condition of the injured persons, the hospital will offer them the nearest hospital that can help the patient and save his life in a minimum of time.

4.2 Preparation and planning of the best routes for emergency ambulances

Traffic congestion is one of the main reasons that create a travel latency for ambulance routing and that is related to the dynamic state of traffic on the ambulance's emergency route. In the case of traffic congestion on the shortest proposed lane, drivers may change directions to other lanes based on new traffic conditions. Since we assume traffic flow (the number of stationary vehicles) and path length as an edge cost in our proposed routing strategy using ant colony optimization (ACO) techniques to build optimal path planning for the ambulance. The number of stationary vehicles on each side is calculated according to the amount of traffic flow on each side received from pre-installed sensors on routes that are centralized on the road traffic center. In our system, we consider the problem as a highly connected graph (N, R) made from a set of GPS positions of ambulances and accidents considered as nodes and routes (edges) that represent the lines or routes between these intersections. To find the optimal route for emergency ambulances, each route is characterized by a cost that combines the length of the route (the distance from the route) and the estimated traffic flow on that route (Figure 5).

5 The principle of the basic optimization of ant colonies

In the real world, ants are able to search for the shortest path between their anthills to food sources. [14], [30] by exploring the real environment based on and relying on a chemical substance called pheromone, which is released on the pathways taken by ants. The proposed Ant Colony System (Ant Colony System) algorithm is a procedure [15], [30] used to simulate the behavior of real ants by producing a new mechanism for artificial ants that extends the same behavior of real ants with the addition of several

attributes that helps the system to converge to optimal solutions in minimum time. The different ACO algorithms depend on the nature of the problem used to solve it, so the parameters of each algorithm are based on what the user needs to look for and the nature of the environment used to apply the ACO technique. L'ACS is based on pre-defined steps defined as follows:

- **Description of the graph:** the artificial entities will move between a set of discrete points in a discrete space. Since the problem posed, by the ACS algorithm (Figure 2) is considered a discrete problem and it can be modeled by a graph built from nodes and edges.
- **Preparation of artificial ants:** a set of ants is distributed over the nodes declared in the proposed graph, the number of ants increases the collection of solutions generated. Since each ant is considered a candidate solution for the given problem.
- **Performing a calculation probability:** ants are based on a probabilistic transition to move between vertices, which is based on two main parameters, are the pheromone and a heuristic argument, which defines the nature of the optimized problem.

Algorithm 1: AS pseudocode algorithm

Input: $N_{nodes}, R_{routes}, A_{ants}$
Result: $best_{route}$

```

1 foreach  $A_i$  in  $A_{ants}$  do
2   foreach  $N_i$  in  $N_{nodes}$  do
3     // Apply a probabilistic
4     transition based on pheromone
5     and heuristic parameter to
6     select best next move and add
7     this node to CR: collection of
8     node
9      $n = select\_next\_node()$  ;
10     $C_n = n$  ;
11     $new\_ph = update\_pheromone()$  ;
12  end
13  // Add constructed path to list of
14  proposed routes
15   $best_{route} = C_n$  ;
16 end

```

Fig. 2. The ACS algorithm

6 Distributed strategy for the routing of emergency ambulances

The proposed solution will define a distributed environment by creating a set of intelligent agents that come together to find the optimal and fastest paths for the emergency ambulance routing problem (Figure 3). It starts by receiving an emergency call from an anonymous user, then the SCA detects the GPS coordinate of the incoming call, which represents the starting node and sends it to the AM agent to have the list of the ambulances closest to the accident site. The shortest path will be assigned to the RA, it will create a set of intelligent agents that will extend the same behavior of real

ants to find the shortest path by implementing the ACS algorithm. When the optimal path is found by the RA, it will be shared with the SCA to start the scenario of routing the ambulance to the position of the accident.

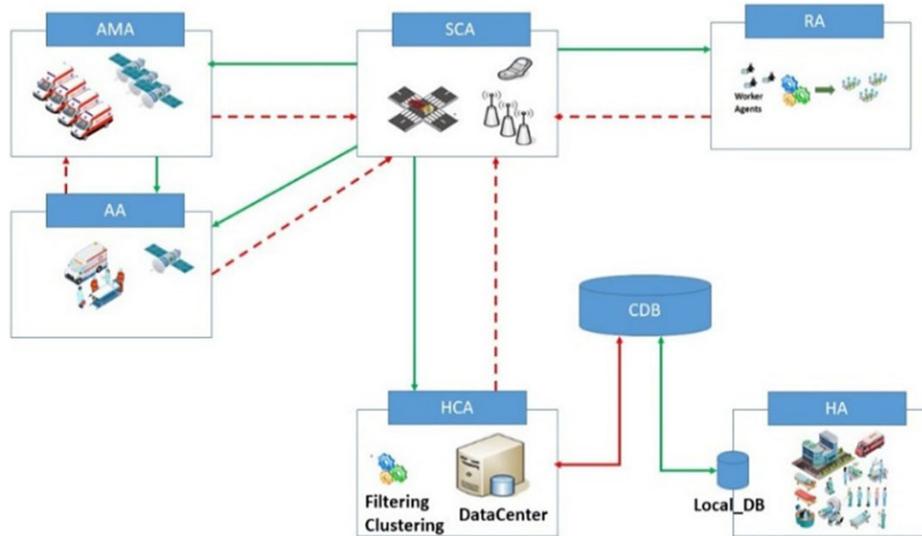


Fig. 3. Decentralized architecture to address the problem of ambulance vehicle routing based on a multi-agent framework

This section describes the distributed approach used to solve the ambulance routing problem, it is based on a set of agents namely:

- **Shore Control Agent SCA**
- **Ambulance Monitoring Agent AMA**
- **Routing Agent RA**
- **Ambulance Agent AA**
- **Hospital Agent HA**
- **Hospital Center Agent HCA**

In the next section, we will describe the work of each agent in our distributed architecture.

6.1 Shore control agent (SCA)

Our system will reserve an agent who will interpret the calls and emergency requests to do the necessary to allocate an ambulance for each accident (Figure 4). The role of this agent is to retrieve the location of the accident and initiate a search process to find the nearest ambulances at the position of the accident. To do this work, an exchange process is initiated between the SCA and the ambulance monitoring agent (AMA) who will provide us with the ambulances inside the accident area. Subsequently, the SCA

will contact the vehicle routing agent (RA) to ask him to prepare the best paths between the current position of each ambulance and the position of the accident using the ACO algorithm. At the end of this optimization process, our RA will offer us a list of journeys between (ambulance, accident), these journeys are calculated in terms of distance and road traffic. The SCA will choose the best among them taking into consideration the total distance of the journey and the time required to reach the position of the accident. Then the best route will be sent to the ambulance, which will take care of this accident. Another process is assigned to this agent, including retrieving the accident condition analysis report. In this report are found a set of key words, each one related to a medical term and has a meaning which will be used to identify the appropriate hospital which can respond to the urgent demand of the accident as quickly as possible. These key-words will be shared with a central HCA agent who controls the hospitals and who is able to consult a centralized database that centralizes all information about the hospitals that can accept and treat this accident. After identifying the hospital that can meet our needs, the SCA will make another call to the RA to ask him to prepare a new route for us between the ambulance position and this hospital. Moreover, the best route will be selected and communicated with the ambulance to guide the ambulance to this hospital as quickly as possible and in the best conditions.

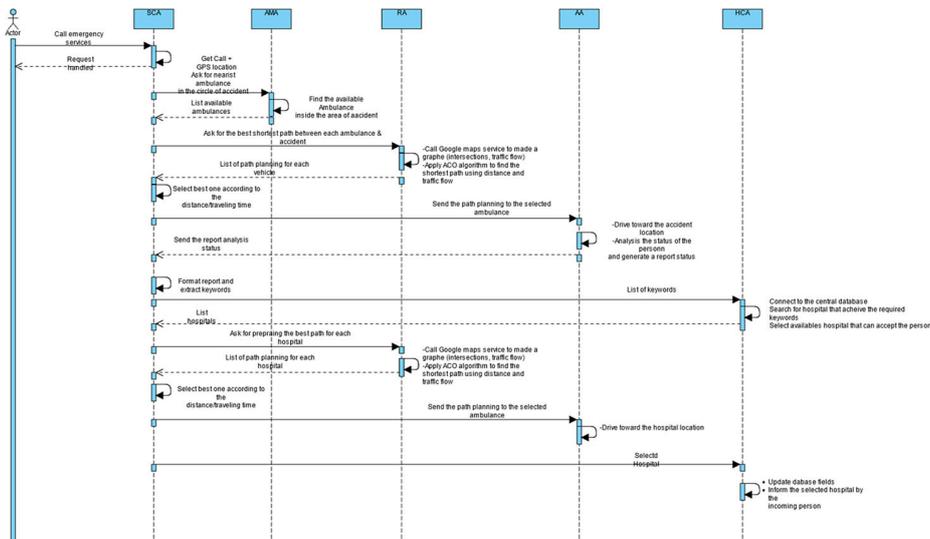


Fig. 4. Communication flow model for emergency ambulance routing

6.2 Agent monitoring ambulance (AMA)

In our case, we assume that each ambulance equipped with a geolocation system through a GPS beacon that shares with us a set of information such as:

- The current position of the ambulance
- Ambulance speed
- The amount of fuel in the tank
- CAN-Bus information that can be used in the ambulance selection process

For the selection of the best ambulances that can respond to the request of the SCA agent, the latter uses the position of the accident as the center of a circle and subsequently launches a geolocation process to identify all the ambulances which are inside this circle (Figure 5), knowing that the radius of our circle is configurable by the system and its default value is 5km. After the identification of these ambulances, the AMA agent shares this collection with the SCA agent to propose all the ambulances that can interpret this accident as soon as possible.

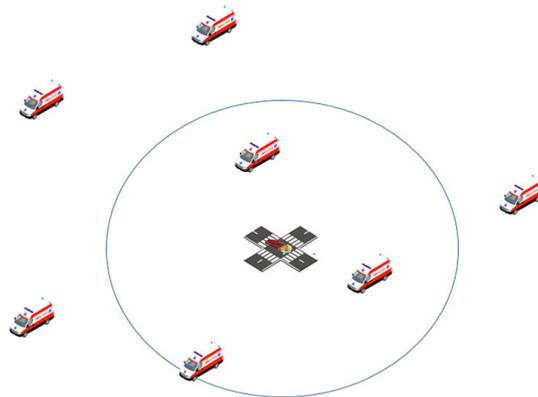


Fig. 5. Identification of the closest ambulances to the accident position

6.3 Routing agent (RA)

This agent uses the Ant Colony System (ACS) algorithm to generate the optimal path for the ambulance. First, it manages the start node (actual position of the ambulance) and the end node (position of the accident) and generates a map that contains all routes and nodes using the Google Maps API. The generated map is considered as a graph, in our case the cost of the edges it refers to the combination of the length of the route (distance) and the traffic flow on this route. Then, the RA creates a set of artificial ants called worker agents (WA) (Figure 6). Each agent will generate a path according to the given graph and the state of the road, after each iteration, a new path is proposed by the (WA) and it is sent to the RA to select the best of them.

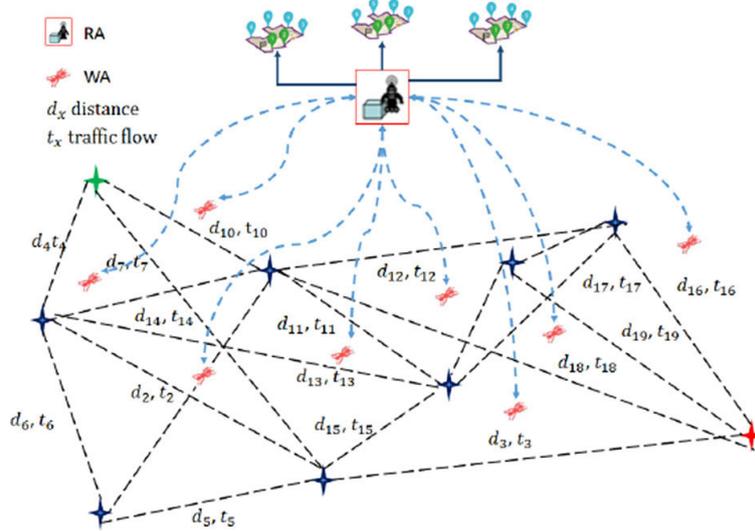


Fig. 6. Distributed interaction model to find optimal routes for an ambulance

When a WA_m is at node n_i , next node n_j will be selected according to a probability transition rule [10], [18] defined as follows (Eq. 2):

$$p_{ij}^m(it) = \frac{\omega_{ij}(it)^\alpha * n_{ij}(it)^\beta * \gamma_{ij}(it)^\delta}{\sum_{l \in N} \omega_{il}(it)^\alpha * n_{il}(it)^\beta * \gamma_{il}(it)^\delta} \quad (2)$$

Or,

- ω_{ij} is the quantity of pheromone deposited on the road
- α is a constraint to control the influence of the pheromone
- n_{ij} is a heuristic function that defines the cost of the route (i, j) . In our case, the cost will be defined as follows: $n_{ij} = 1/d_{ij}$. Where d_{ij} is the distance (km) between the two nodes (i, j)
- β is a constraint parameter to control the influence of distance
- γ_{ij} is a heuristic function representing the density of the traffic flow (travel time) on the edge (i, j) . The $\gamma_{ij} = 1/tf_{ij}$, where tf_{ij} is the density of the traffic flow on the edge (i, j)
- δ represents the influence of traffic flow
- N is the set of unvisited nodes

When a WA finishes its tour, it will be able to deposit a specific amount of pheromone on the edges already visited. The amount of pheromone on each edge is updated according to formula (3). Updating the pheromone will increase the amount of pheromone at the edges followed by the majority of WA. The new amount of pheromone will be used by the new WA to track the better edges that have a higher level of pheromone. The next equation describes how the WA calculates the amount of pheromone that will be applied to the visited roads:

$$\omega_{ij} = (1 - \rho) \cdot \omega_{ij} + \Delta\omega_{ij} \tag{3}$$

$$\Delta\omega_{ij} = \begin{cases} \frac{1}{T_m} & \text{if the } WA_m \text{ use the route } (i, j). \\ 0 & \text{Otherwise} \end{cases}$$

Hence, the parameter ρ represents the evaporation rate between 0 and 1. This parameter is used to decrease the amount of pheromone on the edges less used by ants. T_m is the duration of the tour created by the WA. The amount of newly deposited pheromone is inversely proportional to the total length of the tour found by the WA. The routes with the highest amount of pheromones are considered to be the shortest paths in the given search space. Once all of the WA have completed their rounds, the RA will be able to evaluate the proposed routes and select some of the best preliminary routes. On these routes, we will apply a global update of the pheromones in which the RA adds more pheromones $\Delta\omega_{ij}^{best}$ on the edges included in the previously selected solutions. This update will be done according to the following formula:

$$\omega_{ij} = (1 - \rho) \cdot \omega_{ij} + \Delta\omega_{ij} + \Delta\omega_{ij}^{best} \tag{4}$$

Finally, when the stop condition is met (in our case, we use a maximum number of iterations or the user can stop the running system), the best paths will be sent to the SCA (Figure 7) and all worker agents will be ready to generate a new path according to the new input parameters (traffic state, new input routes, etc.).

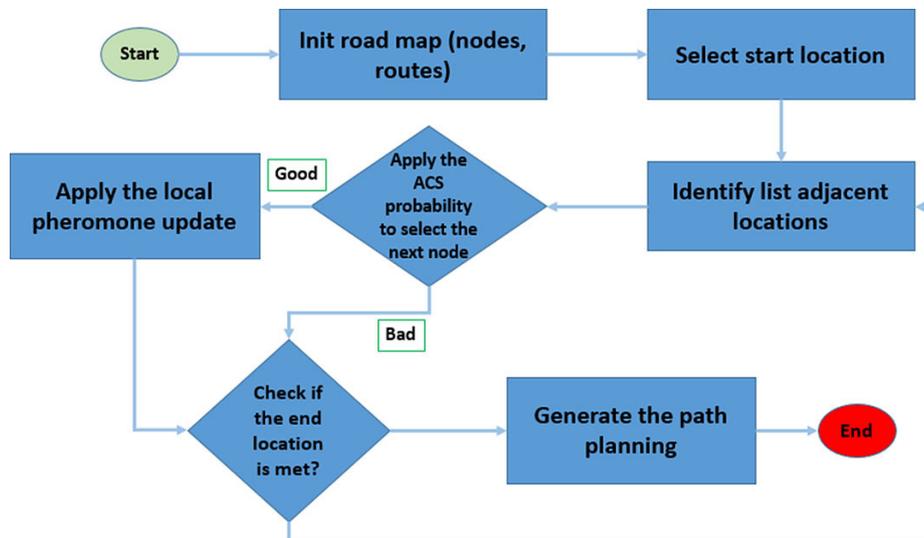


Fig. 7. Flow diagram for ambulance routing

6.4 Ambulance agent (AA)

This agent monitors the status of the ambulance by sending the information provided from the vehicle's RA. This agent will communicate directly with the SCA and give it a set of information such as the real-time position of the ambulance, its current speed. This agent follows the path suggested by the SCA and attempts to guide the ambulance to the patient's location. On the other hand, the AA assesses the state of the accident and sends a report describing in detail the state of the patients. (For example, a 23 year old has a fracture in his left hand, etc.). This information will be useful for the SCA to select and find the optimal hospital that will help this patient immediately.

6.5 Hospital agent

In our architecture, we are in communication with a set of hospital information systems, which must provide us with information on the availability of services, doctors and necessary equipment within the hospital in real time. For this, we thought of making a system of recommendation of interoperability between these hospital information systems using a centralized database which brings together all the information on each hospital. To guarantee interoperability between hospital information systems, we have developed a new communication and information sharing mechanism based on KAFKA topics. In this mechanism, we have information producers and a consumer.

- The producers in our case are the different hospitals. The systems of these hospitals can be developed by any program language (Java Spring Boot, PHP...), this language just needs to support the KAFKA library to produce the message in the topic. The producers produce updated information (Name of the hospital, available service, available doctor) in a topic that we have named Topic_Hospital_Information. Producers provide information in a unified format that is respected by all stakeholders (producer and consumer) of the topic. This unified format is considered as a communication contract between the participants of the topic which is standardized according to a template (Figure 8). The interest of this standardization is to make our system indexable as well as to facilitate the information retrieval and data processing phase. Once the information is put on the topic, a consumer is always listening to consume the information. In our case, we used a Spring Boot micro-service. Once the message is consumed by the micro-service, this micro-service processes to ensure the consistency of the information shared. When the information is properly verified, the micro-service registers the new information in a centralized database CDB. We have adopted a NoSql Cassandra type database DB solution. This database will allow us to use the services provided by this system as well as to optimize the process of inserting and updating data because we are working with a real-time system. Also, to minimize the process of consulting and retrieving information as quickly as possible. If the information processed by the micro-service is not consistent, it will record an error message in the server logs to keep traceability.

```
code_hospital : H001
adresse : Hay amal , zone 3, casablanca
latitude : -7.5667777
longitude : 33.456666
▼ equipments [1]
  ▼ 0 {5}
    code : RD0001
    label : Radio XXX
    disponibility : available
    date_update : 2021-06-06 23:23:08
    nbr_waiting : 3
  ▼ Services [1]
    ▼ 0 {6}
      code : R001
      label : Reanimation
      disponibility : available
      nbr_waiting : 0
      nbr_places : 5
      ► doctors [1]
  ▼ Keywords [2]
    0 : pole-cardiovasculaire
    1 : Chirurgie cardiovasculaire
```

Fig. 8. Example json template

6.6 Center hospital agent

This agent takes the request from the SCA agent which contains an analysis report on the state of the accident. This report is made up of a set of keywords and descriptions to facilitate the process of classification and filtration at the central database level. Sometimes we have several hospitals that can meet our need. For this, our system will offer the SCA agent these hospitals to choose the best among them according to the distance of the journey and the time necessary to reach the hospital according to the traffic conditions. After selecting the best hospital, the CHA agent informs the medical staff within the hospital by the arrival of a new case in order to prepare the necessary services and the medical teams who will treat the patient (Figure 9).

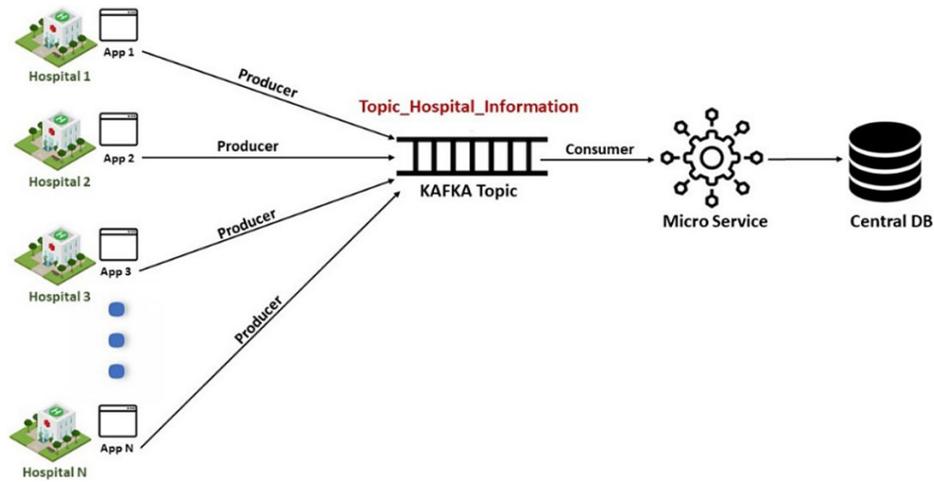


Fig. 9. Hospital information systems interoperability diagram

When paramedics arrive at the accident, they assess the patient’s condition and make a preliminary observation by describing his situation using a set of keywords, these words will be interpreted by the HCA. The latter performs a search on the central database CDB to identify all the hospitals that are able to respond and manage this situation according to the keywords entered. The CBD is updated in real time by other hospitals. According to the list of available hospitals, our system, following the process of finding the shortest route, recommends the hospital closest to the place of the accident, on the other hand, it sends the optimal route that will help and guide the ambulance to recover the wounded to this hospital with minimum travel time (Figure 10).

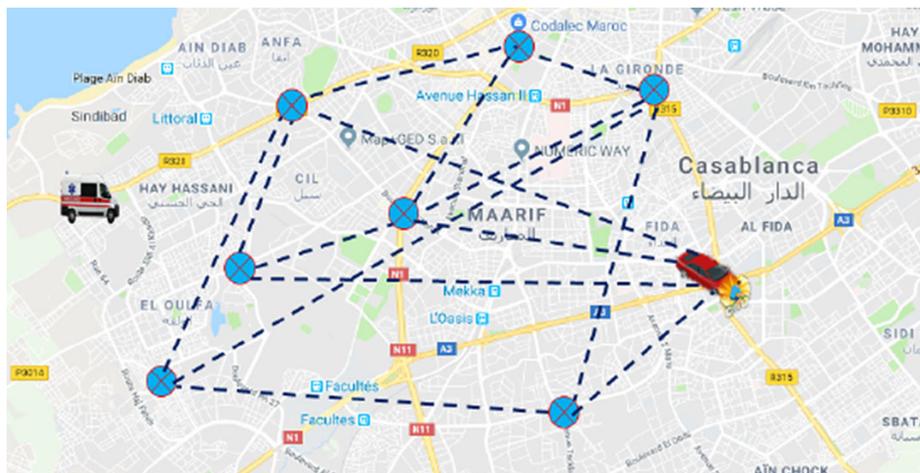


Fig. 10. Road network for transporting emergency ambulances

7 Analysis of the simulation results

To assess the effectiveness of the proposed strategy for the emergency ambulance routing problem, we compare our results with one of the most useful meta-heuristic algorithms is algorithm A* [17], [31]. This algorithm is used to find good feedback for vehicle routing issues and find the shortest path in a graph. The evolution of the history of shortest path techniques was traced back to 1968 and continues to expand depending on the nature of the problems used to solve them. One of the most famous used to find the shortest path was named as the A* algorithm which can calculate the shortest path like Dijkstra's strategy, but the A* is better than Dijkstra at finding good results in a minimum of time for complex problems. Algorithm A* uses heuristic parameters in its strategy to calculate the best path. These heuristic parameters are used to find optimal solutions based on a multiple objective function that has characterized the given problem such as minimizing the total distance from the trip by changing the state of certain constraints in our problem model.

Algorithm A* is able to generate and find good results for the path finding problem, and can achieve excellent running time as compared to the traditional algorithm in the vehicle routing problem. In addition, the A* can be used to find the optimal route between a starting node and a destination node based on a set of parameters such as distance and traffic flow conditions. In our case, the A* algorithm will be implemented to select the optimal path for the ambulance from their stopping station to the accident site and from the accident site to the hospital. The strategy of this algorithm is to skip roads that are very distant or expensive. Model A* can be visualized as follows:

$$f(n) = c(n) + e(n) \quad (5)$$

- n : a specific node on the graph
- $f(n)$: the optimal cost path (shortest line) from the current location to n
- $c(n)$: total cost to reach the next node n from the original node
- $e(n)$: calculated cost of moving node n to destination node

Algorithm A* is based on the minimum $c(n)$ summer $e(n)$ to select the next node n to visit. This process is repeated until you reach the target node. Thus, both the cost of the route from the start to the node n and the estimated cost of the current node n until the last destination are used to select the next best move.

In this section, we present the implementation details of our distributed strategy for emergency vehicle routing. We use the Google Maps Street Data API to generate our map (graph) which will be used to find the optimal route for the ambulance. The data retrieved from this API will be used as input data for our system, we extract the coordinates of the service locations from the GPS which represent the nodes and routes which define the paths between all available nodes. Each node will be characterized by its latitude and longitude and the cost of each route is defined by the distance and traffic flow on each edge. To get the traffic flow, we can call the Google Traffic API or by calling the central route to collect data from the sensor installed on the road. The road network of our problem will be considered as a larger fully connected graph building

from nodes and edges in Java Entity. We implement our proposed solution using a JADE (Java Agent Development Environment) multi-agent framework in which each agent will extend the same behavior of real ants. The Java language will be used to develop our solution by implementing the Java entity-based algorithm (ACS) and we use certain specifications to run our solution on an appropriate environment as follows:

- Processor : Intel® Xeon® CPU E5-1650 v2 @ 3.50GHz, 4 cores
- Kernel and CPU: Linux 4.13.8-1.el7.elrepo.x86_64 on x86_64
- Operating system: CentOS Linux 7.4.1708
- Apache Webserver: 2.4
- Java Version: 1.8
- Google Maps Javascript v3.1

The solution we offer is divided into two main containers: backend and frontend. For the backend side, we implement our distributed algorithm by defining the behavior of our RA agent who will calculate and find the optimal path for the ambulance according to the given entry map according to the set of selected constraints. This container has the ability to exchange information with the SCA, when the RA receives a request to find the shortest path, it will use the input parameters such as origin (ambulance location) and destination (position of the accident), data card (Figure 11) which contains a set of nodes and the intersection with several roads. Using this information, the RA will be able to generate the shortest path for this given chart. For the frontend side, we are creating a web application using HTML and JavaScript tag languages to build our front-end platform. The proposed interface will allow users to select the departure and the destination to be reached. The calculated shortest path will be displayed on the map to visualize the different routes that will be used to reach the accident site (Figure 10). On the other hand, the ambulance personnel will be able to see the planning of the available paths and follow them on the screen of the web application not forgetting the available hospitals that can be visited, which they are able to treat the patient and that they are the closest hospitals to the accident location.

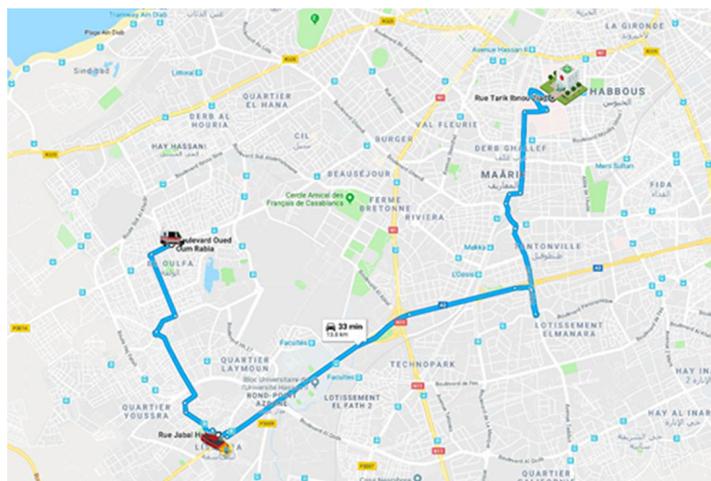


Fig. 11. Optimal planning of the ambulance route to the hospital in the event of an accident

The set of parameters for the execution of our proposed solution is presented in Table 1. We compare our real-time route planning with the shortest path approach proposed by algorithm A*. In both cases, we take into account the traffic flow on the cost of each section (journey between two adjacent intersections). Also for each scenario that represents an emergency call is characterized by a set of intersections that will be explored to find the best route. In our case, we take a scenario of 50 to 600 intersections. We assume that each call is handled only by an ambulance located at the hospital or at a location on the road. First, we compare the performance of the two solutions in terms of total distance travelled (Figure 12), including the distance required to arrive at the accident site and the distance required to return to the nearest hospital. The performance result proves that using the distributed approach, an ambulance needs to spend less time than the A* algorithm-based approach. In this figure, we show feedback on performance in terms of total distance travelled for a set of scenarios.

Table 1. Simulation parameters

Parameters	Specification
Model simulator	Eclipse Luna
Total number of emergency request	50,100,250
Road topology	[50,600]
Parameter α	0.5
Parameter β	0.5
Number of worker ants	30.40.50

Each scenario represents a combined road network with a set of junctions and routes. The distance recovered from both approaches represents the total distance travelled by an ambulance to bring the person from the accident site to the hospital. The performance results show that using the distributed approach, an ambulance needs to spend less time than the A* approaches. We can say that our approach takes advantage of distributed computing based on autonomous agents to find and identify the best paths while taking into account traffic flow and distance. Knowing that the process of collaboration and sharing within our system is asynchronous and goes through a decentralized process to reduce the execution time and maximize the computation flow according to the performance of the work machine.

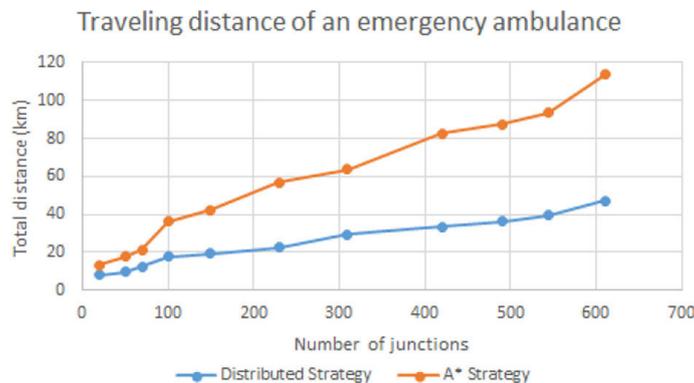


Fig. 12. Comparison of the average travel distance of an emergency ambulance

When we talk about a distributed approach, we must always agree that your solution is able to return the best solutions to us within a reasonable time. For this, we compared the time required to calculate and find the optimal solution for a given request by calculating the time between the processing of the request and the preparation of the best route that will be taken by the ambulance driver. Figure 13 shows that our distributed strategy leverages the parallel process to find and prepare the best solutions in a minimum time, also the strategy is based on a set of agents that they work in parallel to reduce the time consumed to arrive at the optimal solution, on the other hand, in the basic strategy (Algorithm A*) we are based on a sequential iterative process, which consumes more memory and execution time.

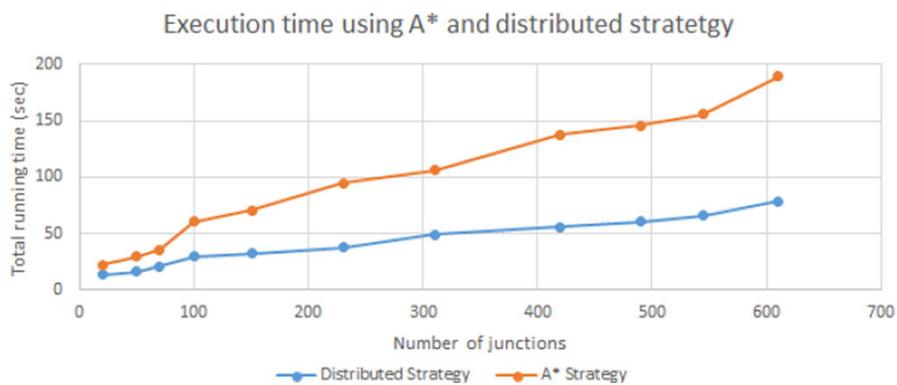


Fig. 13. Comparison of average execution time

8 Conclusion

This work proposes a new architecture for intelligent routing of emergency ambulances by introducing dynamic trajectory planning associated with traffic density using a distributed architecture based on the Ant Colony System (ACS) algorithm combined with the road sensor controller. A recommendation system has also been proposed for collecting information from different hospital information systems and centralizing it on a single database. This CDB will be consulted by the HCA Agent to find the hospital nearest to the emergency ambulance. The main difference between the search engine of this system and other similar approaches is the convergence control of the ACS algorithm based on dynamic information extracted from the road network such as traffic flow and position layout of accidents and hospitals. This approach is modeled by a decentralized strategy in which the search engine is done inside an intelligent agent and the control system is supervised by the ShoreControl agent. It is a distributed approach, traffic flow data is collected from several sensors installed on the road and each emergency vehicle can follow the path proposed by the system to reach its destination in a minimum of time. The main objective of the proposed study is to use it in real cases. The proposed allocation system requires further progress in social and road infrastructure and future studies may include booking on-road routes only to ambulance in the event of an emergency.

9 References

- [1] H. Ibrahim Bulbul, T. Kaya, Y. Tulgar, “Analysis for Status of the Road Accident Occurance and Determination of the Risk of Accident by Machine Learning in Istanbul,” *15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, December 2016. <https://doi.org/10.1109/ICMLA.2016.0075>
- [2] R. Sánchez-Mangas, A. Garcia-Ferrer, A. de Juan, A. Martín Arroyo, “The Probability of Death in Road Traffic Accidents. How Important Is a Quick Medical Response?” *Accident Analysis and Prevention*, Volume 42, Issue 4, July 2010, Pages 1048–1056. <https://doi.org/10.1016/j.aap.2009.12.012>
- [3] P. Sarathi Chakraborty, A. Tiwari, P. Raj Sinha, “Adaptive and Optimized Emergency Vehicle Dispatching Algorithm for Intelligent Traffic Management System,” *Procedia Computer Science*, Volume 57, 2015, Pages 1384–1393. <https://doi.org/10.1016/j.procs.2015.07.454>
- [4] M. Abed Mohammed, M. Khanapi Abd Ghani, R. Ibraheem Hamed, Salama A. Mostafa, M. Sharifuddin Ahmad, D. Ahmed Ibrahim, “Solving Vehicle Routing Problem by Using Improved K-Nearest Neighbor Algorithm for Best Solution,” *Journal of Computational Science*, April 2017. <https://doi.org/10.1016/j.jocs.2017.04.012>
- [5] P. R. Katre, A. Thakare, “Survey on Shortest Path Algorithm for Road Network in Emergency Services,” *2nd International Conference for Convergence in Technology (I2CT)*, Mumbai, 2017. <https://doi.org/10.1109/I2CT.2017.8226158>
- [6] X. Li, Z. Zhao, X. Zhu, T. Wyatt, “Covering Models and Optimization Techniques for Emergency Response Facility Location and Planning: a Review,” *Mathematical Methods of Operations Research*, Volume 74, Issue 3, 2011, Pages 281–310. <https://doi.org/10.1007/s00186-011-0363-4>
- [7] A. Javidaneh, M. Ataee, Ali A. Alesheikh, “Ambulance Routing With Ant Colony Optimization,” *Geospatial Information Systems*, Volume 6, 2010.
- [8] E. Babae Tirkolaee, M. Alinaghian, A. Asghar Rahmani Hosseinabadi, M. Bakhshi Sasi, A. Kumar Sangaiah, “An improved ant colony optimization for the multi-trip Capacitated Arc Routing Problem,” *Computers & Electrical Engineering*, Volume 77, 2019, Pages 457–470. <https://doi.org/10.1016/j.compeleceng.2018.01.040>
- [9] M. Elgarej, K. Mansouri, M. Youssfi, N. Benmoussa, H. Elfazazi, “Distributed Swarm Optimization Modeling for Waste Collection Vehicle Routing Problem,” *International Journal of Advanced Computer Science and Applications (IJACSA)*, January 2017. <https://doi.org/10.14569/IJACSA.2017.080943>
- [10] A. Byrski, E. Swiderska, J. Lasisz, M. Kisiel-Dorohinicki, T. Lenaerts, D. Samson, B. Indurkha, A. Nowé, “Socio-cognitively Inspired Ant Colony Optimization,” *Journal of Computational Science*, Volume 21, 2017, Pages 397–406. <https://doi.org/10.1016/j.jocs.2016.10.010>
- [11] A. Nayyar, R. Singh, “Ant Colony Optimization Computational Swarm Intelligence Technique,” *3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, New Delhi, 2016.
- [12] N. Yadav, V. Yadav, P. Verma, “Role of Evolutionary Algorithms in Software Reliability Optimization,” *International Conference System Modeling & Advancement in Research Trends (SMART)*, Moradabad, 2016. <https://doi.org/10.1109/SYSMART.2016.7894487>
- [13] W. Rao, F. Liu, S. Wang, “An Efficient Two-Objective Hybrid Local Search Algorithm for Solving the Fuel Consumption Vehicle Routing Problem,” *Applied Computational Intelligence and Soft Computing*, 2016. <https://doi.org/10.1155/2016/3713918>

- [14] M. Tomera, "Ant Colony Optimization Algorithm Applied to Ship Steering Control," *Procedia Computer Science*, Volume 35, 2014, Pages 83–92. <https://doi.org/10.1016/j.procs.2014.08.087>
- [15] M. Graf, M. Poy, S. Bischof, R. Dornberger, T. Hanne, "Rescue Path Optimization Using Ant Colony Systems," *IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017. <https://doi.org/10.1109/SSCI.2017.8280873>
- [16] P. Y. Cheong, D. Aggarwal, T. Hanne, R. Dornberger, "Variation of Ant Colony Optimization Parameters for Solving the Travelling Salesman Problem," *IEEE 4th International Conference on Soft Computing & Machine Intelligence (ISCMI)*, 2017. <https://doi.org/10.1109/ISCMI.2017.8279598>
- [17] N. A. Mohd Nordin, N. Kadir, Z. A. Zaharudin, N. A. Nordin, "An Application of the A* Algorithm on the Ambulance Routing," *IEE Colloquium on Humanities, Science and Engineering*, Penang, 2011. <https://doi.org/10.1109/CHUSER.2011.6163858>
- [18] S. Yoshimura, T. Ono, "Multiobjective Optimal Route Finding for Ambulances Using the Genetic Algorithm to Minimize Transport Duration and Blood Pressure Variation of a Patient," *The SICE Annual Conference 2013*, 2013, Pages 2296–2301.
- [19] G. Zhang, "Optimization Research and Exploration of New Rural," *International Journal of Online and Biomedical Engineering (iJOE)*, Volume 17, Issue 3, 2021. <https://doi.org/10.3991/ijoe.v17i03.20695>
- [20] Dai Sheng-Hui, Zhu Xue-Qin, Gui Ying, Xu Hong-Zhen, "An Improved Bayesian Learning Method for Multi-agent System," *International Journal of Online and Biomedical Engineering (iJOE)*, Volume 11, Issue 9, 2015: "Wireless Sensor Network Applications in Engineering". <https://doi.org/10.3991/ijoe.v11i9.5071>
- [21] P. Luca, P. Sernani, A. Claudi, G. Dolcini, G. Biancucci, A. Franco Dragoni, "A Multi-Agent Architecture for Health Information Systems," *Proceeding of International Workshop on Artificial Intelligence and NetMedicine. IJCAI, China*, 2013, Pages 41–50.
- [22] Cardoso, Luciana, Fernando, Marins, Filipe, Portela, Manuel, Santos, António, Abelha José, Machado, "A Multi-agent Platform for Hospital Interoperability," *Springer International Publishing, Ambient Intelligence—Software and Applications, Advances in Intelligent Systems and Computing*, Volume 291, Pages 127–134. https://doi.org/10.1007/978-3-319-07596-9_14
- [23] T. Tlili, M. Harzi, S. Krichen, "Swarm-based Approach for Solving the Ambulance Routing Problem," *Procedia Computer Science*, Volume 112, 2017. Pages 350–357. <https://doi.org/10.1016/j.procs.2017.08.012>
- [24] W. Sulistiadi, S. Nurhidayah, A. Asyary, "Evaluating the Management Information System of Integrated Medical Emergency Care in Batang Regency, Indonesia," *International Journal of Online and Biomedical Engineering (iJOE)*, Volume 16, Issue 7, 2020. <https://doi.org/10.3991/ijoe.v16i07.14725>
- [25] N. A. Mohd Nordin, N. Kadir, Z. A. Zaharudin, N. A. Nordin, "An Application of the A* Algorithm on the Ambulance Routing," *IEEE Colloquium on Humanities, Science and Engineering*, Penang, 2011. <https://doi.org/10.1109/CHUSER.2011.6163858>
- [26] H. Salehinejad, F. Pouladi, S. Talebi, "Intelligent Navigation of Emergency Vehicles," *Developments in E-systems Engineering*, Dubai, 2011. <https://doi.org/10.1109/DeSE.2011.17>
- [27] M. Hosseini, Richard B. Berlin Jr, L. Sha, "Physiology-Aware Rural Ambulance Routing," *The Fifth IEEE International Conference on Healthcare Informatics (ICHI 2017)*, Park City, Utah, 2017. <https://doi.org/10.1109/ICHI.2017.27>
- [28] M. Gendreau, G. Laporte, F. Semet, "A Dynamic Model and Parallel Tabu Search Heuristic for Real-time Ambulance Relocation," *Parallel Computing*, Volume 27, Issue 12, 2001. Pages 1641–1653. [https://doi.org/10.1016/S0167-8191\(01\)00103-X](https://doi.org/10.1016/S0167-8191(01)00103-X)

- [29] C. Vasic, D. Rancic, “Relocation of Resources in Road Transportation Based on the Use of Data From the GPS Based Systems,” *24th Telecommunications Forum (TELFOR), Belgrade, 2016*. <https://doi.org/10.1109/TELFOR.2016.7818713>
- [30] X. Wei, L. Han, L. Hong, “A Modified Ant Colony Algorithm for Traveling Salesman Problem,” *International Journal of Computers, Communications & Control (IJCCC), August 2014*. <https://doi.org/10.15837/ijccc.2014.5.1280>
- [31] D. Li, M. Liu, J. Zhang, E. Cheng, “An Improved A* Algorithm Applicable for Campus Navigation System,” *International Conference on Network and Information Systems for Computers, Wuhan, 2015*. <https://doi.org/10.1109/ICNISC.2015.72>

10 Authors

Meryem Fakhouri Amr is a PhD student at the laboratory SSDIA (Signals, Distributed Systems and Artificial Intelligence) of the Normal School of Technical Education ENSET of Mohammedia, University Hassan II of Casablanca, Morocco. She is currently working as a Systems Administration Engineer. His main interests are the interoperability of information systems, the transformation of MDA models, Web services, business intelligence and ontologies. Email: meryemfakhr@gmail.com

Mouhcine Elgarej obtained her PhD in Computer Science in 2020 at the Normal School of Technical Education ENSET of Mohammedia, University Hassan II of Casablanca, Morocco. Currently he works as a software developer. His main interests are multi-agent systems.

Nezha Benmoussa is a teacher of computer science, project management and research professor at the ENSET Mohammedia, University Hassan II of Casablanca, Morocco. PhD in 2019 at the Faculty of Technical Sciences of Mohammedia. His research is focused on SIAD Decision Support Information Systems.

Khalifa Mansouri is a teacher of computer science and researcher at the Hassan II University of Casablanca, ENSET Institute. His first PhD in 1994 in Mohammed V University of Rabat, HDR in 2011 and second PhD in Hassan II University of Casablanca, Morocco. His research is focused on Real-time modeling systems, information systems, e-Learning systems and modeling, simulation and optimization of industrial systems.

Mohamed Qbadou is a teacher of computer science and researcher at the University Hassan II Casablanca, ENSET Institute. His research is focused on semantic Web, Distributed computing system, Knowledge database system, educational information system and e-learning, Assistive robotics, Natural Language processing. HDR in 2017 at the Hassan II University of Casablanca, Morocco.

Article submitted 2021-07-12. Resubmitted 2021-08-11. Final acceptance 2021-08-12. Final version published as submitted by the authors.