

Semantic Micro-Services Model for Vehicle Routing using Ant Colony Optimization

Asmaa ROUDANE^{*1}, Mohamed YOUSSEFI², Khalifa MANSOURI³

Lab. SSDIA, ENSET, University Hassan II Casablanca, Morocco
Bd. HassanII, Mohammedia, 28820 Morocco

Abstract—In this paper, we propose a new model of vehicle routing based on micro services using the principle of selection and composition of paths. In this model each micro service is responsible for a road resource, a step in the driver's journey, implementing a specific objective of the road course. The micro services are deployed in a cloud architecture in multiple instances according to a system of increase in load and fault tolerance. Drivers requests are sent to a proxy micro-service, with abstract structures of road courses represented by oriented graphs. The proxy micro-service is responsible for analyzing the request to determine the driver's profile and its context in the journey in order to select the micro-service responsible for providing information on the most appropriate road resource. The method of meta-heuristic optimization of road courses, used in the proposed approach, is based on the ant colony algorithm, or we will describe an adaptation of this optimization method to propose to the driver, at each stage of the journey, the most optimal resource to exploit.

Keywords—Road traffic; road traffic management systems; intelligent systems; complex systems; graphs; real-time; optimization methods; micro-services; ant colonies; vehicle routing

I. INTRODUCTION

Information Technologies have evolved in a remarkable way the Information Systems (IS) of the companies. In recent years, Web technology, virtualization of architectures and cloud computing have largely contributed to the evolution of the Information Systems dimension and have forced organizations to restructure themselves to adapt to these technological evolutions. The development of geo-localized mobile systems such as smart phones and miniaturized embedded systems such as Raspberry and Arduino has opened a new area that allows us to interconnect the objects of our universe to scientific applications. Web 3.0 is coming to fruition and the Internet of Things (IOT) has become a reality that Information Systems must take into consideration [1]. In fact, the objects in our environment can be equipped with processing units capable of performing treatments, making decisions while being permanently connected to the Internet and thus providing applications with features that make it possible to exploit the intelligence massively collective. The great mass of data (Big Data) produced by connected objects [2] [3] and mobile applications requires the adoption of a new IS data storage and analysis strategy.

In the field of road traffic, vehicle routing must take advantage of the use of all these new information and communication technologies on the one hand, and all other road traffic management methods on the other hand. Faced

with the massification of users in the road network, the automation and optimization of vehicle routing is an operation that is necessary to minimize the travel time of drivers to keep a maximum flow of traffic and avoid the phenomenon of congestion.

In front of an increasingly open environment, the selection and composition of road resources requires a more efficient strategy for better reuse of traffic data. Micro-services [3], [4], [5] are an architectural style that has emerged in recent years and are best prepared for massively distributed systems. The main idea is to break down a large application into very small, weakly coupled and independent units called "micro-service". Each micro-service must be responsible for a single feature, developed and deployed independently of each other. Based on a lightweight communication mechanism, this style seems to be an opportunity for road traffic management systems where communication between stakeholders is complex.

To help the user to cross the most optimal path to his destination we will exploit the optimization methods. Combinatorial optimization can involve several domains, always with the aim of improving the functioning by means of a sophisticated management of the resources taking into account the temporal and dynamic characteristics of the system [6][7]. In our case we will describe an adaptation of meta-heuristic optimization method by ant colonies to propose to the driver, at each stage of the journey, the most optimal resource to exploit.

The meta-heuristics of ant colony optimization (ACO) was initiated mainly by Marco Dorigo in 1992 [8], for the search of the shortest path in a graph. It was inspired by the behavior of real ants made by Deneubourg and al in 1983 [9]. The first ant colony optimization algorithm that was developed to solve the problem of the commercial traveler is: Ant System (AS) and was introduced by Dorigo and Gambardella in 1997 [10]. The goal was to optimize a route linking a set of cities. Subsequently, several applications of ACO were presented by researchers, including the problem of routing vehicles in 1999 by Bullnheimer and al [11].

II. PRESENTATION OF THE MODEL ARCHITECTURE

Consider a path represented by an oriented graph $G(V_i, E_j)$ where the nodes V_i represent the steps to reach ($i = 0$ to $N-1$, with N represents the total number of nodes) and E_j represent the arcs of the graph defining the different possible paths that the driver can take to reach his destination ($j = 0$ to $M-1$, where M represents the number of arcs of the graph). For each

^{*}Corresponding Author

node i , we associate several possible road resource implementations $R(i, k)$ ($k = 0$ to $H-1$, where H represents the number of possible resources associated with a node). A road resource can be either a normal step, a rest area, a service station or a control point. During his journey, the driver should explore a road resource for each node according to his personal choices or according to a recommendation system.

Over time, according to the feedback of the drivers and the results obtained by the users in the various control points of the course, each resource $R(i, k)$ of each node i is assigned a score S_i and each arcs E_j a weight W_j reflecting the relevance of the path. The system should also be able to determine, over time, the optimal route that provides the best route for drivers. Fig. 1 shows an example of a path composed of 8 nodes and 9 arcs.

To reach a destination $V(i)$, we define several implementations of road resources $R(i, k)$ possible. As said previously, a road resource can be a normal step in the route, a rest area, a service station, a control point, or an aggregate of a set of resources. In this model, we propose to expose each of the road resource implementations using a micro-service $MS-R(i, k)$ deployed in a cloud architecture. To handle the problem of increase in load and fault tolerance, multiple instances of the same micro-service $MS-R(i, k)$ can be deployed thus defining a massively distributed architecture.

To facilitate the orchestration process, all micro-services should register and publish the description of the exposed resource in a special micro-service named $MS-D$ representing the semantic system directory. This description includes both the functional and non-functional properties of micro-services. This is an important element in the selection and composition of services. We also propose to define a micro-service specially designed to centralize and expose the configuration of all micro-services of the platform.

In this model, all driver requests are sent to a proxy micro-service named $MS-PROXY$, specifically designed to handle the orchestration process by invoking the appropriate micro-service to serve the requested resource. For this, the proxy service consults the registration and discovery service to find the location of the most appropriate micro-service and will be asked to respond to the contextual request. In the case where the proxy service needs to communicate the same information to several micro-services, it should use a more efficient messaging system based on the AMQP protocol by using one of the broker implementing this protocol as Rabbit MQ. The schema of Fig. 2 shows a technical architecture showing all the micro services offered in this model inspired from a related work [12].

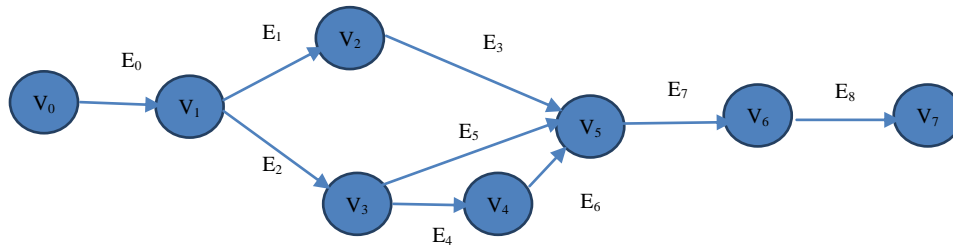


Fig. 1. Representation of a Path by an Oriented Graph.

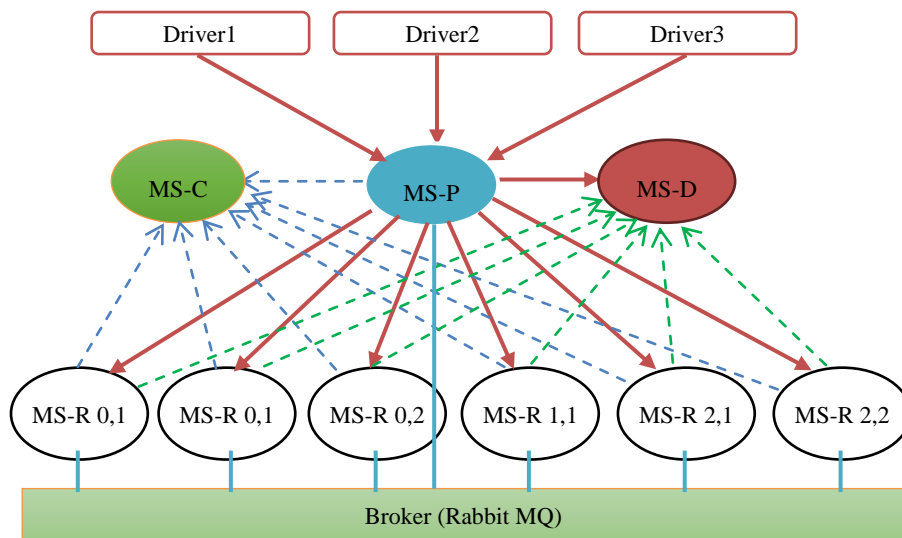


Fig. 2. Micro-Services Architecture of the Model.

III. ROAD RESOURCE MODEL

A road resource is a structure that contains the contents of a fragment of the route. This could be a normal step in the trip, a rest area, a service station, or a control point. Generally, a road resource is an aggregate of an element of the route, service station and control point. The level of granularity of the resource depends on the context of the journey and has a considerable influence on the size and efficiency of the system. A finer level of granularity makes it possible to follow the driver more closely, thus enabling the system to effectively evaluate the drivers' journeys and to target their problems with great precision and also to facilitate their orientation. However the number of micro-services that will be deployed in the system would be too important. Hence, the need to deploy the system in a cloud environment [13]. On the other hand, a large level of granularity of road resources reduces the size of the platform, but also reduces the level of efficiency of the system. Indeed, with large basic road resources, it is more difficult to follow the driver more closely, which makes the control of the course more difficult.

To include in our model the different possibilities, we propose to use the composite pattern to structure an elementary road resource (Fig. 3). Each road resource is described by a set of functional or non-functional properties. The functional properties make it possible to describe the semantics of the resource, its coordinates, its type, its level of recommendation, etc. Non-functional properties are related to the technical aspects of the resource such as safety, cost, weight, recommended flow etc.

IV. DESCRIPTION OF MICRO-SERVICES IN THE ARCHITECTURE

A. Micro-Service of Road Resources

This micro-service is the central element of the model, because it's it, in collaboration with its counterparts, which exposes the state of the road network to the driver. At the moment of its start, this micro-service first connects to the configuration service to search its configuration and then to the registration service to record its location and publish its description in the dedicated directory for.

Technically, it has its own layers of data, business, and web:

- The data access layer allows the structuring and persistence of its basic content in light storage systems such as micro-DBMS or simple XML files.
- The business layer defines for the micro service the necessary processing to perform to manage the content of the supported resource. Generally, the processes of this layer evolve within the transactions which make it possible to guarantee the coherence of the data recorded during the interactions with the drivers.
- The micro service web layer exposes all business layer processes via a REST API or other remote access components such as a SOAP-based Web service. Access to this layer is restricted mainly to the Proxy micro-service that manages access to drivers in a more efficient manner.

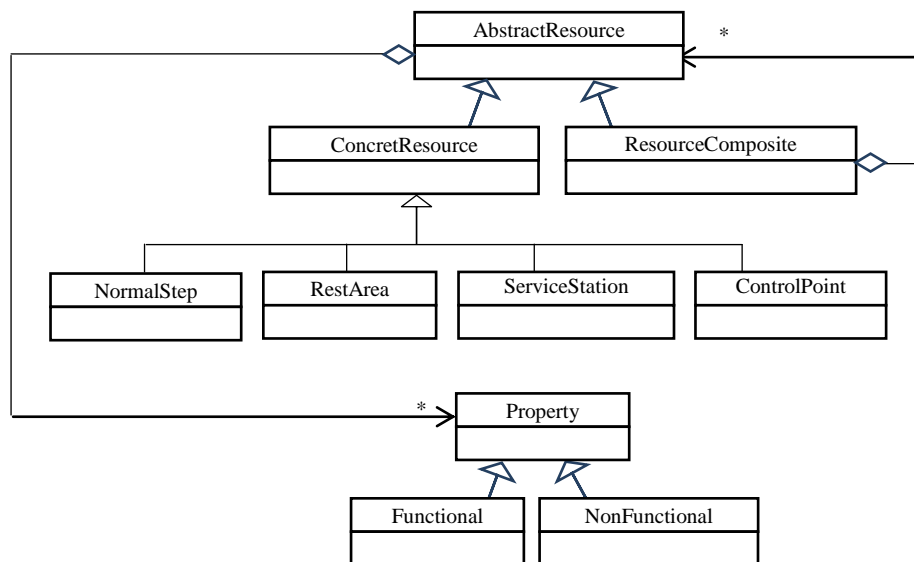


Fig. 3. Structure of a Road Resource.

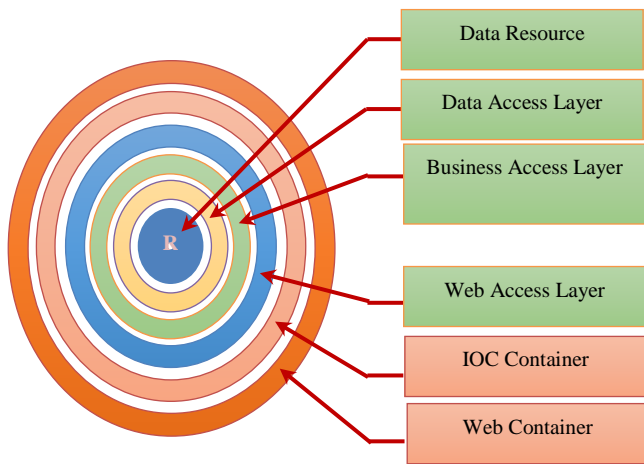


Fig. 4. Structure of a Road Resource Micro-Service.

The micro-service evolves in a container which ensures the principle of the inversion of control based on a micro-Framework IOC. The latter makes it possible to deal implicitly and to easily inject the technical aspects into the micro-service. This is the case for security aspects, transaction management, persistence management, logging, monitoring, hot configuration, cloud services, migration, and so on. To expose the micro-service features via http and SOAP protocols, a web container is also used in the architecture. Fig. 4 shows the different software layers of this micro-service inspired from a related word [14].

B. Proxy Micro-Service

As noted in the previous section, all driver requests are sent to the Proxy micro-service. The latter finds in the request the identifier of the route plan and that of the requested road resource. The proxy service requests the registration service to provide a list of deployed micro services that provide the content (status) of the road resource in question. Using a load balancing algorithm combined with the scoring system relating to the relevance of its resources, the proxy designates the appropriate micro-service and redirects it to retrieve the resource to be sent to the driver.

Following the feedback from the driver, the score of this resource is updated. This score will be used to calculate the arc weight linking this resource and that of the previous node in the path of the driver's journey. This step is part of an algorithm for optimizing pathways by meta-heuristics using the principle of ant colonies. This optimization system, which will be presented in the following section, provides the driver with the best path among those defined in the course plan. The communication diagram of Fig. 5 shows a diagram of interactions between the driver, the micro-service proxy, the Discovery micro-service, two instances of the micro-service exhibiting the same resource implementation relative to the contextual objective and another instance of the micro-service exposing another implementation of the resource of the same contextual objective.

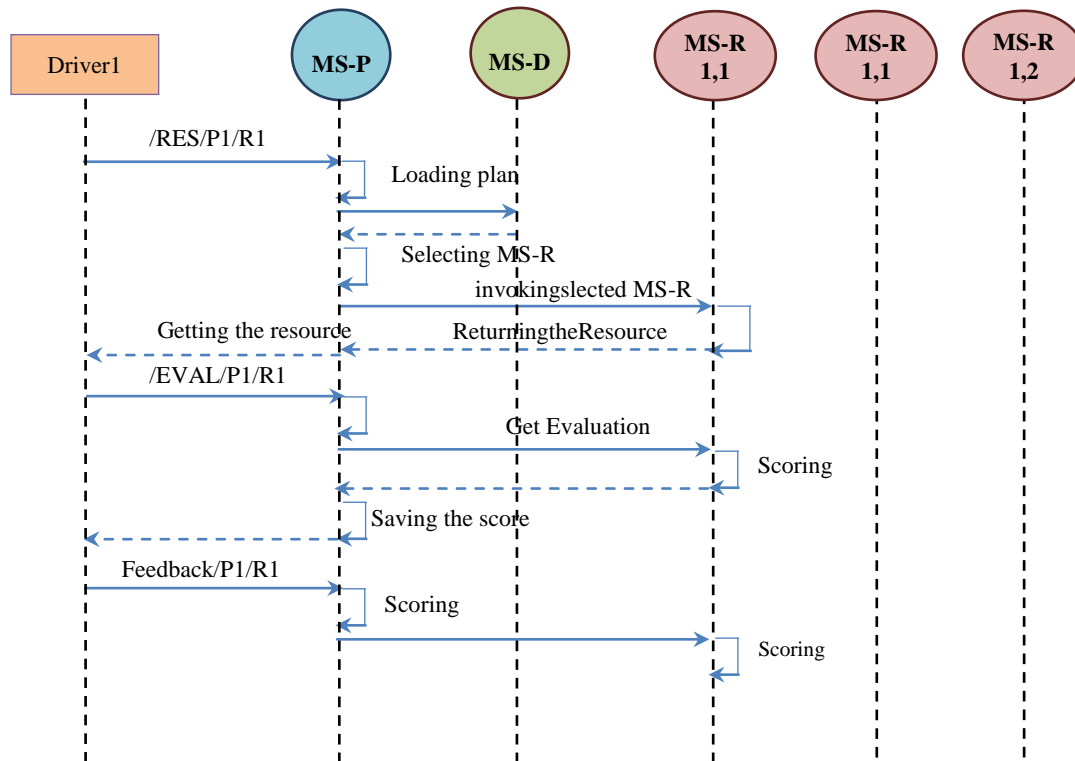


Fig. 5. Interaction Diagram Showing the Functioning of Micro-Services of the Model.

V. ALGORITHM FOR OPTIMIZING ROAD COURSE BY ANT COLONIES

Ant colony algorithms are algorithms inspired by the behavior of ants, or other species forming a super organism, and which constitute a family of meta-heuristics of optimization. Initially proposed by Marco Dorigo et al. [15], in the 1990s for the search of optimal paths in a graph, the first algorithm is inspired by the behavior of ants looking for a path between their colony and a food source. The original idea has since diversified to solve a broader class of problems and several algorithms have emerged, inspired by various aspects of ant behavior. The idea comes from observing the exploitation of food resources in ants. Indeed, these, although having individually limited cognitive abilities, are able by exploiting the collective intelligence, to find the shortest way between a food source and their nest.

A model explaining this behavior is as follows:

- An ant (called "scout") runs more or less randomly around the colony.
- If it discovers a food source, it returns more or less directly to the nest, leaving on its way a trail of pheromones.
- These pheromones are attractive, ants passing nearby will tend to follow, more or less directly, this track.
- Returning to the nest, these same ants will strengthen the track.
- If two tracks are possible to reach the same food source, the one that is the shortest will be, at the same time, traveled by more ants than the long track.
- The short track will be more and more reinforced, and therefore more and more attractive.
- The long track, it will eventually disappear, pheromones are volatile by evaporation phenomenon.
- In the long term, all the ants thus determined and "choose" the shortest track.

In our case, we consider our plane of paths represented by an oriented graph of Fig. 1. Each node represents an element of the road network and each arc a possible route between two elements. Each of these arcs has its weight W initialized by

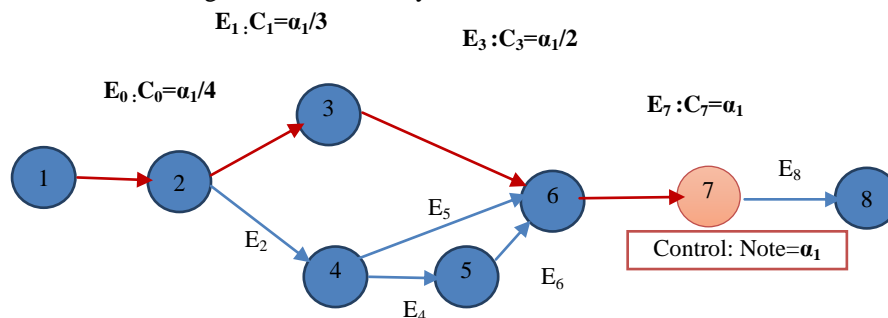


Fig. 6. Principle of Retro-Propagation of Pheromones.

the team of managers of the road traffic management system, to mark a relative importance between the relevance of several possible paths from the same origin. Each driver who travels the graph is represented by an "ant", a micro-agent, who navigates on the underlying graph, by taking a path E_j to arrive at a node V_E proposing a control of the acquired, the ant releases pheromones C whose quantity depends on the note α obtained at the end of this control.

To reflect the phenomenon of retro-propagation, pheromones are not simply released on the arc that led the ant to the current node but on the last n last arcs that the ant followed. This is done to reflect the fact that to reach its destination all the characteristics of the roads that lead to it count. Of course, this influence diminishes with time and space: the further the node is removed in the history of the path $P(a)$ of the ant, the less it is important. For this to be taken into account, the amount of pheromone deposited decreases as retro-propagation advances. In our implementation of this algorithm, we propose that the amount of pheromone that would be added in the path E_k in the history $P(a)$ is the quantity α divided by a number $d(k)$ which represents the distance separating the node V_E subject destination, at the origin node of the arc E_k .

$$C_k = C_k + \frac{\alpha}{d_k} \tag{1}$$

Fig. 6 shows an example of retro-programming of pheromones for the case of a driver who has passed through a control at node 7 and obtains a note α_1 . To arrive at this node, he followed the path $P=\{E_0, E_1, E_3, E_7\}$. The amounts of pheromones deposited in arcs E_7, E_3, E_1 and E_0 of this course would be respectively $\alpha_1, \alpha_1/2, \alpha_1/3$ et $\alpha_1/4$.

The fact that biological pheromones evaporate over time is extremely important because it allows the ant colony to rely on constantly updated information. In our artificial system, it is important to implement a form of evaporation to avoid that the system remains "stuck" in a local optimum and to open the door to the expected characteristics of dynamic adaptability:

$$C_t = \tau^x C_{t-1} \tag{2}$$

Equation 2 gives the form of evaporation for pheromones. τ , the evaporation rate, is a key setting parameter of the system. The period of evaporation, x which says at which intervals the evaporation is calculated is a system constant. Typically, $\tau=0.999$ and $x=1$ day.

In addition to the results of the controls carried out by the drivers, the road resources will be subjected to an appreciation of the users. Each road resource is therefore assigned a rating β from each driver. This score will be multiplied by a credibility factor μ associated with each user by the system and the management team. As in the previous case of the note α assigned by the system to the drivers, the note relating to the appreciation of the drivers will be translated in quantity of pheromones A placed on all the arcs which led the users towards this resource by respecting the principle of retro-propagation of equation 1:

$$A_k = A_k + \frac{\mu \cdot \beta}{d_k} \quad (3)$$

The phenomenon of evaporation of pheromones A is established in the same way as equation 2:

$$A_t = \tau^x A_{t-1} \quad (4)$$

The values W , C and A are so-called collective factors, they concern all drivers who will use the information they contain to build their path. W is exogenous, it is brought by the team of managers to the community drivers/ants. C and A are endogenous, they are created by individuals for the service of the community: immersed intelligence, emerging culture. The system must also be able to take into account individual factors, specific to each driver, in order to achieve the compromise that is sought between the individual, the community and the environment.

The individual factors that can be taken into account are numerous (preferences, excellence, history, etc.). The idea is to allow the system to propose solutions according to the individual himself. In our case, we will consider the historical factor H .

The History Factor H carries the information on the previously visited nodes in the history of the driver (vehicle). Since this is an individual factor, there is an H value per driver per node. The default value of H is chosen to 1. When a node is visited by a driver, the corresponding value H is multiplied by a factor h inversely proportional to the score obtained. For example we can choose the following values:

$h = 0.25$ if the score is between 75% and 100%,

$h = 0.5$, if the score is between 50% and 75%,

$h = 0.75$ if the score is between 25% and 50%

$h = 0.95$ if the score is less than 25%.

The role of H is to decrease the probability that the node already visited is proposed again. Of course, if the node was the seat of a very bad score ($h = 0.95$) it will be more quickly proposed again than if there is a very good score ($h = 0.25$).

Over time, of course, drivers forget what they saw. This is why H tends to return naturally to 1. This "anti-evaporation" is described by the equation that models this phenomenon of anti-evaporation of memory pheromones.

$$H_t = H_{t-1} \left(1 + \frac{1-H_{t-1}}{H_{t-1}} \frac{1-e^{\gamma x}}{1+e^{\gamma x}} \right) \quad (5)$$

γ is a time constant that regulates the speed of the phenomenon. It must be adjusted to match the volatility of the

drivers' memory. It is therefore a constant that must be the subject of a dialogue with the managers. Equations 6 and 7 allow an easy calibration of γ . Equations 4 and 5 show how to calculate the value of γ as a function of the values of H at times t and $t-1$ and the duration x during which a driver could possibly forget a resource already explored before.

$$\gamma = \frac{1}{x} \ln \left(\frac{1+\tau}{1-\tau} \right) \quad (6)$$

$$\tau = \frac{H_t - H_{t-1}}{1 - H_{t-1}} \quad (7)$$

All the factors previously described (W , C , A and H) are unified by a function called "fitness", by analogy with the literature of genetic algorithms. Equation 8 shows the value expression of the "fitness" of the arc E_{ij} leading from the node i to the node j for the individual a . H is the memory factor, W is the weight proposed by the system, C is the control pheromone concentration and A is the pheromone concentration of the driver's feedback. The ω_H , ω_w , ω_c and ω_A are parameters that make it possible to give more or less importance to one factor compared to others. This function measures the excellence of a given arc, its "desirability", and will condition, through a selection procedure, the probability that the arc considered will be suggested to drivers.

$$f(a, E_{i,j}) = \omega_H H(j, a) (\omega_w W + \omega_c C + \omega_A A) \quad (8)$$

We can say that an arc is desirable when:

- He is encouraged by the team of road traffic managers (W high).
- He sees good ratings attributed by the system to drivers in control points (high C).
- He sees good appreciation ratings attributed by drivers to road resources (high A).
- The node at which it ends has never been visited by the driver or was forgotten by him (H close to 1).

When a driver validates a node, it is advisable to choose, among the arcs that come out of this node, the one that is most suitable to borrow, this is the role of the selection procedure. This is where the fitness measures are used. A selection procedure will choose an arc randomly but all the more so if the fitness of the arc is high. We will see the "effective" arcs become predominant but not exclusively, there will be room for chance and exploration. This part of chance is a crucial feature of the selection procedure, it is called selective pressure or s . The greater is s , the more the draw is guided by the fitness function and the stronger arcs will tend to dominate the weak ones.

VI. APPLICATION AND RESULTS

In this part, we propose the results of a simulation corresponding to an application implemented at the level of the proxy micro-service, which consists in randomly launching HTTP clients, each representing a driver. Each driver takes his route from one node to another in the route plan graph. We have chosen an example of a part of the course whose structure represented by Fig. 7.

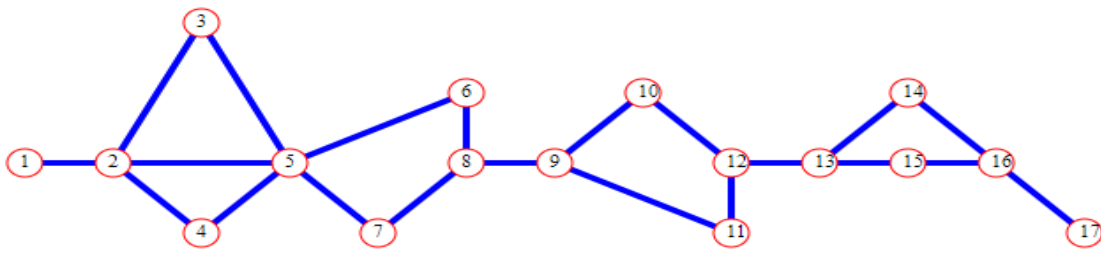


Fig. 7. Course Structure Proposed in this Application.

This course plan is an oriented graph that consists of the following nodes:

- Nodes 1, 2, 5, 9 and 13 represent normal steps in the path.
- Nodes 3, 6, 10 and 14 represent service stations.
- Nodes 4, 7, 11 and 15 represent rest areas.
- Nodes 8, 12, 16 and 17 represent control points.

In this proposal, the weight of a path proposed by managers is simply the distance between two nodes. When there are several links leaving a node, we have chosen the values of the relevance proposed by the managers: W, allowing more recommending the passage by paths which have the shortest distances. This is the case of the links (2 => 3), (2 => 4) and (2 => 5) which respectively have the values of W: 111.80, 70.71 and 100. Which means that the team managers recommend to borrow, in order of preference, a rest area (4) or a normal stage (5) or a service station (3).

When passing through a control point, a random note C is generated, representing the amount of pheromones left on its path (equation 1). Otherwise the driver leaves an appreciation of the selected road resource representing the amount of pheromones A left on its course (equation 4) by activating the evaporation procedure before proceeding to the next node.

Fig. 8, 9 and 10, respectively show the representation of the evolution curves of the parameters W, C, A and Fitness as the drivers follow their journeys.

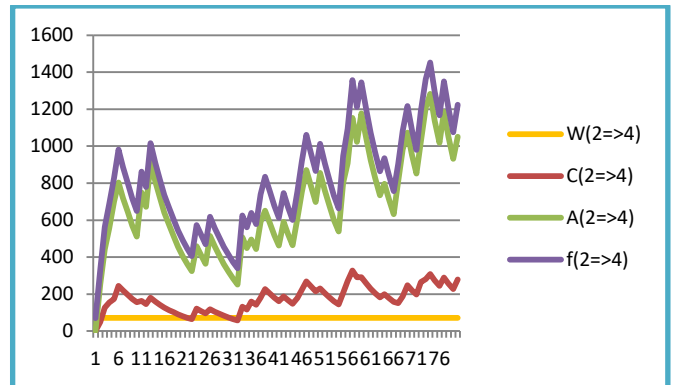


Fig. 9. Evolution of W, C, A and Fitness Values for the Link (2 => 4).

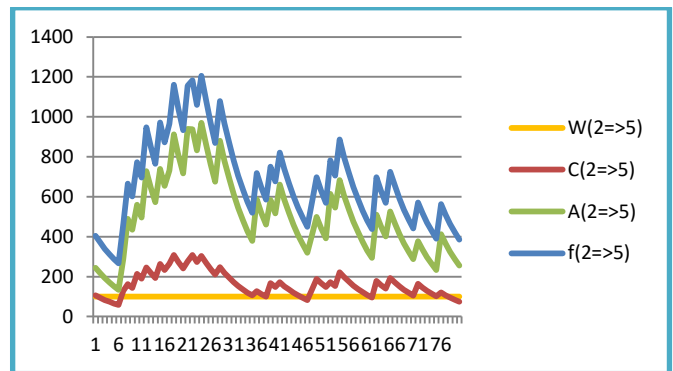


Fig. 10. Evolution of W, C, A and Fitness Values for the Link (2 => 5).

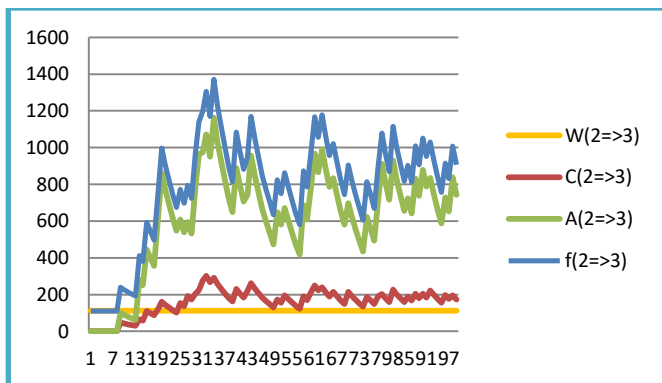


Fig. 8. Evolution of W, C, A and Fitness Values for the Link (2 => 3).

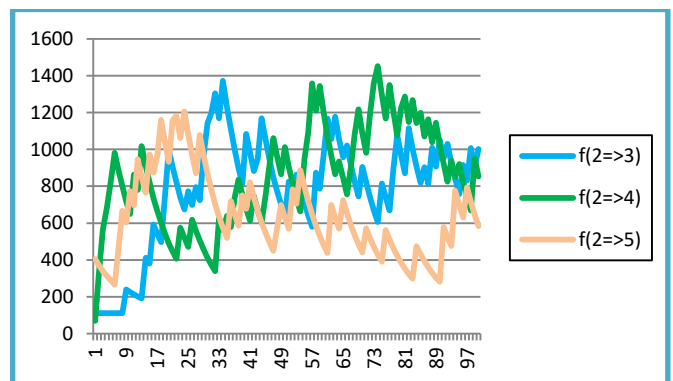


Fig. 11. Fitness Comparisons of the Three Links (2 => 3), (2 => 4) and (2 => 5).

REFERENCES

Fig. 11 represents a comparison between the fitness of the 3 links (2 => 3), (2 => 4) and (2 => 5). We note that most of the time that the link (2 => 4) is more borrowed, reinforcing the recommendation of the team of managers by taking the path that has the shortest distance. As in the system, there is a lot of randomness, dominance can change direction. This is the case of the driver N° 34 which makes that the link (2 => 3) becomes momentarily more dominant. This is still the case of the driver 24 who leaves the fitness function of the arc (2 => 5) more dominant instantly when it was not recommended by the team of managers. This reveals other parameters that are difficult to take into consideration, such as random excellence, driver serenity and the relevance of assessments in control points. Despite all this, the model makes it possible to obtain, in the end, interesting results thus resisting the random phenomena mentioned.

VII. CONCLUSION

The model proposed in this contribution revolves around optimizing agents aiming to improve the vehicles routing. The approach is based on the ant colony algorithm to improve road traffic management by helping to select the best possible path for a given destination. Ant behavior is characterized by its autonomy, distributed operation and self-organizing capabilities to form systems capable of performing very complex tasks. For this reason, our model is inspired by heuristic research that has involved many criteria in the objective function on a distributed multi-agent architecture in which artificial agents are created by analogy with social insects. The proposed model can provide a new way to improve traffic conditions and optimize the use of resources to provide the driver with the best guidance to their destination. In future work we plan to propose and implement other optimization algorithms based on other parameters of the traffic environment including safety and cost of the road (not only the distance of the journey), and this to be able to select the most optimal path to a destination by adding other factors according to the preferences of the user. We also aim to combine the ACO (ant colony) algorithm and the Q-Learning reinforcement learning algorithm to improve the modeling of the optimizers agents of the vehicle routing system.

- [1] H. Rolf, Weber, "Internet of Things - Need for a New Legal Environment?", *Computer Law & Security Review*, pp. 522-527, 2009.
- [2] Yunchuan Sun, Houbing Song, Antonio J. Jara, Rongfang Bie, *Internet of Things and Big Data Analytics for Smart and Connected Communities*, *IEEE Access* (Vol: 4), pp. 766 - 773, February 2016.
- [3] Ejaz Ahmed et al, *The role of big data analytics in Internet of Things*, *Computer Networks* Volume 129, Part 2, pp. 459-471, December 2017.
- [4] D. Namiot, M. Sneys-Snepe, "On micro-services architecture", *International Journal of Open Information Technologies*, vol. 2, no. 9, 2014.
- [5] D. Malavalli, S. Sathappan, "Scalable microservice based architecture for enabling dmtf profiles", *Network and Service Management (CNSM) 2015 11th International Conference on*, pp. 428-432, Nov 2015.
- [6] D. Jaramillo, D. V. Nguyen, R. Smart, "Leveraging microservices architecture by using docker technology", *SoutheastCon 2016*, pp. 1-5, March 2016.
- [7] C.H. PAPANITRIOU, K. STEIGLITZ, *Combinatorial optimization - algorithms and complexity*. Prentice Hall, 1982.
- [8] C.C. RIBEIRO, N. MACULAN (Eds.), *Applications of combinatorial optimization*. *Annals of Operations Research* 50, 1994.
- [9] Dorigo, M. (1992). *Optimization, learning and natural algorithms*. PhD Thesis, Politecnico di Milano.
- [10] Deneubourg, J. L., Pasteels, J. M., & Verhaeghe, J. C. (1983). Probabilistic behaviour in ants: a strategy of errors?. *Journal of Theoretical biology*, 105(2), 259--271.
- [11] Dorigo, M., & Gambardella, L. M. (1997). Ant colony system: a cooperative learning approach to the traveling salesman problem. *IEEE Transactions on evolutionary computation*, 1(1), 53--66.
- [12] Bullnheimer, B., Hartl, R. F., & Strauss, C. (1999). An improved ant System algorithm for the vehicle Routing Problem. *Annals of operations research*, 89, 319--328.
- [13] Ahmed ESSAYAH, Mohamed Youssfi, Omar Bouattane, Khalifa Mansouri and Elhoccin Illoussamen, "QoS-based Semantic Micro Services Discovery and Composition using ACO Algorithm" *International Journal of Advanced Computer Science and Applications(IJACSA)*, 10(6), 2019.
- [14] FAREH, Mohamed El-kabir (2015) Une approche basée agents pour l'allocation des ressources dans le Cloud Computing. Masters thesis, Université Mohamed Khider - Biskra.
- [15] A. Coloni, M. Dorigo et V. Maniezzo, *Distributed Optimization by Ant Colonies*, actes de la première conférence européenne sur la vie artificielle, Paris, France, Elsevier Publishing, 134-142, 1991.