

The Proposal of a Distributed Algorithm for Solving the Multiple Constraints Parking Problem

Khaoula Hassoune¹, Wafaa Dachry², Fouad Moutaouakkil¹, Hicham Medromi¹

¹Systems Architecture Team ,Laboratory of Research in Engineering, Hassan II University, ENSEM Casablanca, Morocco

²Laboratory of Mechanical Engineering, Industrial Management and Innovation Hassan I University, FST Settat, Morocco

Abstract—The parking problem in big cities has become one of the key causes of the city traffic congestion, driver frustration and air pollution. So to avoid these problems, parking monitoring is an important solution. Recently many new technologies have been developed that allows vehicle drivers to effectively find the free parking places in the city but these systems still limited because they don't take into consideration road networks constraints. In this paper, We design a distributed system that will help drivers to find the optimal route between their positions and an indoor parking in the city taking into consideration a set of constraints such as (distance, traffic, amount of fuel in the car, available places in te parking, and parking cost). We propose a distributed technique based on multi objective Ant Colony Optimisation (ACO). The proposed method aim to manage multi objective parking problem in real time using the behavior of real ants and multi agent systems to decrease the traffic flow and to find the optimal route for drivers.

Keywords—ant colony optimization(ACO); multi-agent system; parking monitoring; intelligent systems

I. INTRODUCTION

Cities noticed that their drivers had real problems to find a parking space easily especially during peak hours, the difficulty roots from not knowing where the parking spaces are available at the given time. Even if this is known, many vehicles may pursue a small number of parking spaces which in turn leads to traffic congestion. So researchers are recently turned to apply intelligent transportation technologies for management of parking area by designing and implementation of smart parking systems that allows vehicle drivers to effectively find the free parking places. The main idea is to reduce traffic and time searching for a parking place.

Many systems are developed for parking but each system is based on different Technologies . To control the parking problem we need to use an intelligent system that can be adaptative with the environment and give us the optimal path to reach an indoor parking place taking into consideration multiple constraints.

Parking problem is situated in several researches based on different technologies and algorithms. Some of these systems are based on intelligent systems. In this domain, we have systems that are based on internet of things and cloud computing [1] [2] [3]. and others which are based on Artificial intelligence like fuzzy logic [4] and neural networks [5].

Ant colony optimization (ACO) [6] is a very powerful optimization heuristic for combinatorial optimization prob-

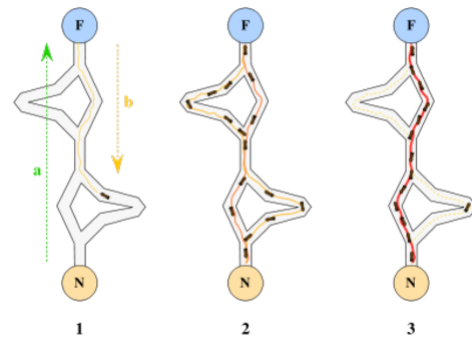


Fig. 1. The behavior of real ants

lems, Vehicle Routing algorithms [7] [8], and NP-complete problems. Ant Colony Optimization (ACO) is inspired by ants behavior to find the shortest paths between their colony and the source of food. Ants manage to discover the optimal path in an organized and distributed manner (Figure 1). In an ACO algorithm ants are programmed agents to find an optimal combination of elements in a given set of utility functions. The key ingredient in ACO is pheromones which are deposited chemicals by ants and their concentration traces a map of trajectories. Paths that contain a highest concentration of pheromones represent better trajectories. In this paper, we propose a mathematical model to decrease the time searching for an indoor parking taking into consideration a set of constraints (distance, traffic, amount of fuel in the car, available places and parking cost) . With this approach, we propose a new method to optimize parking timing based on a distributed multi-objective ACO algorithm and multi agent systems.

The outline of this work is as follows. In the next section, we define related works done to solve this type of problems. In Section 3, we describe the multi-objetive problem and how can be modeled as a graph. The Mathematical model for solving the multi objective parking problem is proposed in Section 4 that aims to find the optimal solution for drivers. Section 5 explains the distributed multi objective ACO based on multi-agent system for parking problem. Section 6 summarizes the perspectives. Finally, the conclusion is given in the last section.

II. PREVIOUS WORKS

In the literature, many researches about smart parking systems have been presented. These researches have used different information technologies like: wireless sensor network, internet of things, cloud Computing systems, mobile application, and artificial intelligence techniques.

The authors in [9] [10] present the design and implementation of a smart parking system based on wireless sensor networks that allow vehicle drivers to find the free parking places.

In [11] the author presents a wireless system for locating parking spots remotely via smartphone. This system automates the process of locating an available parking spot and paying for it.

Authors in [12] have proposed a scalable and low cost car parking framework (CFP) based on the integration of networked sensors and RFID technologies. These framework include driver guidance, automatic payment, parking lot retrieval, security and vandalism detection.

In others studies the authors have choose to design an automatic smart parking using internet of things which enables the user to find the nearest parking area and the available slot in that area [1] [2] [3].

In [13], an intelligent parking guidance algorithm using dynamic city parking conditions has been proposed. Simulation results were provided to validate the effectiveness of the proposed algorithm.

Samaras [4] proposed a fuzzy logic system based on the energy efficiency of the city 's parking lots. The wireless sensor network sends the actual data to generate fuzzy rules in the study.

Wang [14] has developed a based reservation smart parking that allows drivers to find and reserve the vacant parking places. Cai [15] proposed a navigation system based on the parking sensor network using a Dijkstra optimization algorithm to obtain the optimal parking route. . Zheng [5] has developed a prediction system for parking problem using some machine learning methods such as artificial neural network, support vector machines, and regression) using real data acquired from two cities like Melbourne and San Francisco. Mejri [16] proposed a multi-purpose intelligent parking approach based on a mata heuristic to optimize parking location using real data.

We notice that the guidance systems proposed by researches dont provide an optimal multi objective solution to find an indoor parking with available places. We design a distributed system that will help drivers to find the optimal route between their positions and an indoor parking in the city taking into consideration a set of constraints such as (distance, traffic, amount of fuel in the car, available places and parking cost). The proposed method is based on multi objective ACO and intelligent agents .

III. PROBLEM DEFINITION

In this case, the parking problem can be modelled as a graph with distributed drivers, service stations and parkings

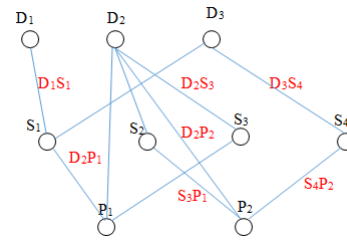


Fig. 2. The problem graph

where $D = \{D_1, D_2, D_3, \dots, D_m\}$ is The set of drivers searching for parking, $P = \{P_1, P_2, P_3, P_4, \dots, P_n\}$ is the set of indoor parkings and $S_f = \{S_1, S_2, S_3, \dots, S_p\}$ the set of service stations.

Let G be the graph (Figure 2) with vertices in $D \cup P \cup S$ (called nodes) and the edges (arcs) in E defined as follows: $E = \{D_i S_j / 1 \leq i \leq m \text{ and } 1 \leq j \leq p\} \cup \{S_i P_j / 1 \leq i \leq p \text{ and } 1 \leq j \leq n\} \cup \{D_i P_j / 1 \leq i \leq m \text{ and } 1 \leq j \leq n\}$. We denote $D_i S_j$ the distance between the vertice D_i and S_j (between the driver D_i and the service station S_j), $S_i P_j$ the distance between the vertice S_i and the parking P_j (between the service station S_i and the parking P_j) and $D_i P_j$ the distance between the vertice D_i and P_j (between the driver D_i and the parking P_j).

A given route may contain a service station. the objective of the problem is to find the optimal path for each driver searching for a parking place by considering the following constraints:

- A Driver searches for an indoor parking with available places.
- Every route starts from the driver and finishes in a parking.
- The amount of fuel in the vehicle must allow the driver to reach at least one service station and a car park.
- Other parameters such us (traffic, distance, Available places, cost of parking, and amount of fuel) need to be taken into consideration in our optimization problem.

IV. ANT COLONY OPTIMISATION BASED MATHEMATICAL MODEL

In an ACO algorithm ants are programmed agents to find an optimal combination of elements in a given set of utility functions. The key ingredient in ACO is pheromones which are deposited chemicals by ants and their concentration traces a map of trajectories. Paths that contain a highest concentration of pheromones represent better trajectories. ACO is among heuristic methods that use distributed optimization concepts and pheromone concentration to solve combinatorial optimization problems [17]. The Ant Colony System Algorithm is divided into three main steps:

- All ants will construct their tours based on a probabilistic transition which is related to the amount of pheromone in their paths. After the construction of the trajectories, the ants will update the pheromone amount on each arc visited.

- The system will find the optimal path provided by the ant colony and chooses the optimal solution for each driver.
- In the end, the best ants update the overall pheromone on the paths that are part of the best solution.

The whole procedure repeats until the end of a number of iterations. The ACO (Ant Colony Optimization) algorithm for parking problem starts with a group of ants. These ants work together simultaneously to build routes. Each ant will build a complete circuit based on a given parking map. The construction of the route can be described as follows: First, the artificial ants (an intelligent developed agent) start from the driver 's location 's and select the next node to visit. the ant has two possibilities: The first is to select a gas station location to visit from the list of available gas station locations to reach a parking lot with available spaces and the second is to select a parking location to visit from the list of available parking spaces. When the ant reaches the parking 's position, then the system will restart the procedure of searching for another optimal path for an indoor parking with available places. In this version of ACO for parking problem, each ant needs to create a driver route that visits an indoor parking. Each ant select the next neighbor to visit using a probabilistic transition [18], if the ant is in a driver's position we have two possibilities (1) (2) :

$$p^k(D_i, S_j) = \frac{\tau^\alpha(D_i, S_j) \prod_{l \in \widetilde{P}_1} \eta^{\alpha_l}(D_i, S_j)}{\sum_{\beta \in [1, p]} \tau^\alpha(D_i, S_\beta) \prod_{l \in \widetilde{P}_1} \eta^{\alpha_l}(D_i, S_\beta)} \quad (1)$$

$$p^k(D_i, P_j) = \frac{\tau^\alpha(D_i, P_j) \prod_{l \in \widetilde{P}} \eta^{\alpha_l}(D_i, P_j)}{\sum_{\beta \in [1, n]} \tau^\alpha(D_i, P_\beta) \prod_{l \in \widetilde{P}} \eta^{\alpha_l}(D_i, P_\beta)} \quad (2)$$

where:

$\widetilde{P}_1 = \{ \text{distance, traffic, amount of fuel in the car} \}$ is the set of parameters of edges $D_i S_j$.

$\widetilde{P} = \{ \text{distance, traffic, available places, parking cost, amount of fuel in the car} \}$ is the set of parameters of edges $D_i P_j$.

$\tau^\alpha(D_i, S_j)$: value of pheromone trail on edge from D_i to S_j

$\tau^\alpha(D_i, P_j)$: value of pheromone trail on edge from D_i to P_j

α : coefficient that controls importance level of level of pheromone

$\eta^{\alpha_l}(D_i, S_j)$: value of cost functions for parameter l for edge from D_i to S_j

$\eta^{\alpha_l}(D_i, P_j)$: value of cost functions for parameter l for edge from D_i to P_j

α_l : coefficient that controls importance level of parameter l

The cost functions for all parameters for edge $D_i S_j$ are :

$$\eta^d(D_i, S_j) = \frac{1}{D_i S_j} \quad (3)$$

The equation (3) presents the cost function for distance $D_i S_j$ between the driver D_i and the service station S_j .

$$\eta^d(D_i, P_j) = \frac{1}{D_i P_j} \quad (4)$$

The equation (4) presents the cost function for distance $D_i S_j$ between the driver D_i and the parking P_j .

$$\eta^t(D_i, S_j) = \frac{1}{\text{tr}(D_i, S_j)} \quad (5)$$

The equation (5) presents the cost function for traffic where $\text{tr}(D_i, S_j)$ is the measured traffic flow from the driver D_i to the service station S_j .

$$\eta^t(D_i, P_j) = \frac{1}{\text{tr}(D_i, P_j)} \quad (6)$$

The equation (6) presents the cost function for traffic where $\text{tr}(D_i, P_j)$ is the measured traffic flow from the driver D_i to the parking P_j .

$$\gamma(D_i) = \frac{\text{fuel}}{\text{distance}} \quad (7)$$

The equation (7) presents the amount of fuel consumed in litres by the driver D_i per km.

$$\eta^f(D_i, S_j) = \text{sup}(F(D_i) - \gamma(D_i) * (D_i S_j), 0) = \eta^f(D_i, S_{j0}) \quad (8)$$

$\eta^f(D_i, S_{j0})$ is the minimal fuel cost to reach the service station S_{j0} and Where $F(D_i)$ is the amount of fuel in the vehicle of the driver D_i .

$$\eta^f(D_i, P_j) = \text{sup}(F(D_i) - \gamma(D_i) * (D_i P_j), 0) = \eta^f(D_i, P_{j0}) \quad (9)$$

$\eta^f(D_i, P_{j0})$ is the minimal fuel cost to reach the parking P_{j0} and Where $F(D_i)$ is the amount of fuel in the vehicle of the driver D_i .

$$\eta^c(D_i, P_j) = \frac{1}{\text{cost}(P_j)} \quad (10)$$

The equation (10) presents the cost function for parking cost $\text{cost}(P_j)$ in the parking P_j .

$$\eta^v(D_i, P_j) = \text{Card}(P_{jv}) \quad (11)$$

where $P_j = P_{jv} \cup P_{j\widetilde{v}}$ and $P_{jv} \cap P_{j\widetilde{v}} = \emptyset$

P_{jv} is the set of available places in the parking P_j and $P_{j\widetilde{v}}$ is the set of non available places in the parking P_j . $\text{Card}(P_{jv})$ is the number of elements of the set (P_{jv}) .

When the ant reach a Service station 's position we use the probabilistic transition to select the next parking to visit :

$$p^k(S_i, P_j) = \frac{\tau^\alpha(S_i, P_j) \prod_{l \in \widetilde{P}} \eta^{\alpha_l}(D_i, P_j)}{\sum_{\beta \in [1, n]} \tau^\alpha(S_i, P_\beta) \prod_{l \in \widetilde{P}} \eta^{\alpha_l}(S_i, P_\beta)} \quad (12)$$

where:

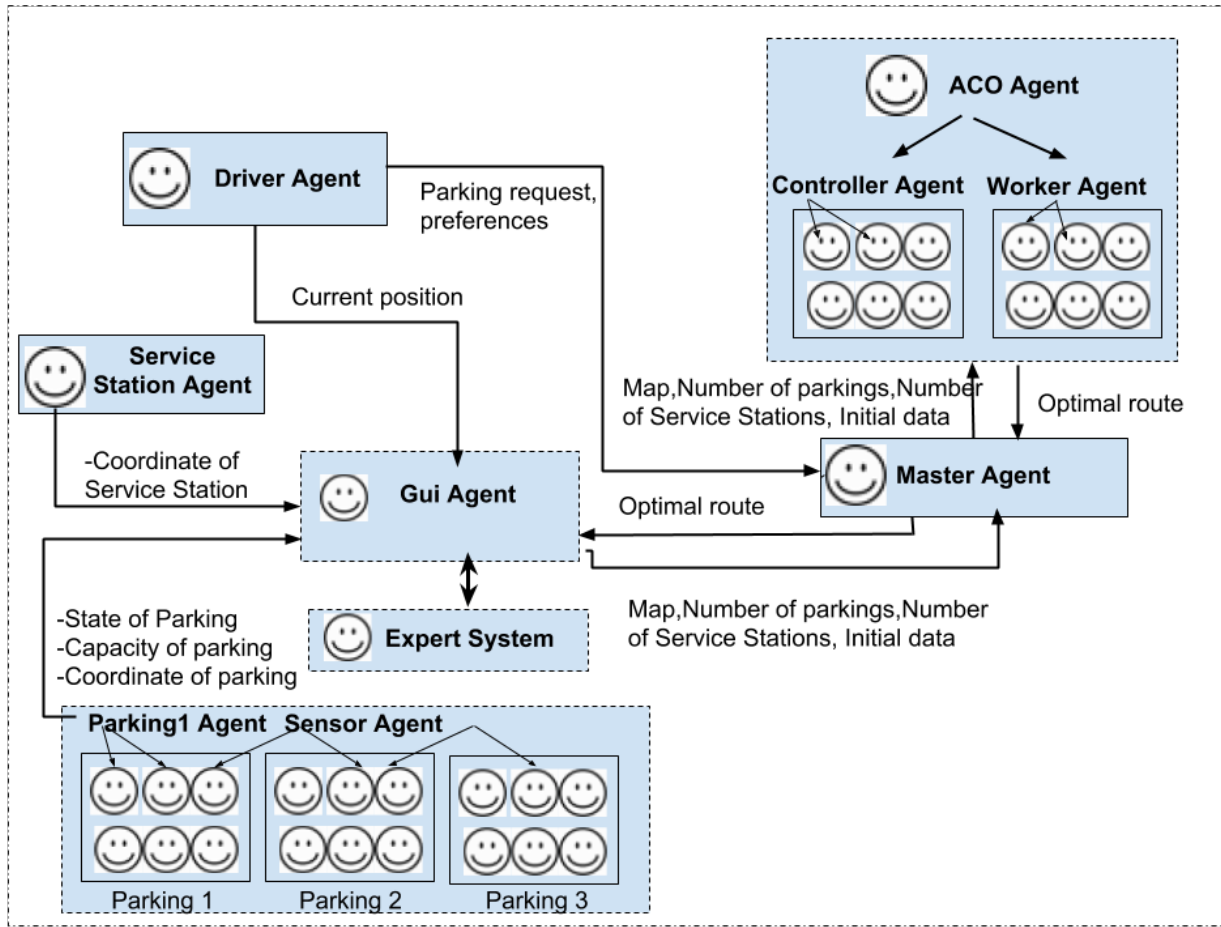


Fig. 3. Distributed Architecture for Parking

$\tilde{P} = \{ \text{distance, traffic, available places, parking cost, amount of fuel in the car} \}$ is the set of parameters of edges $D_i P_j$.

$\tau^\alpha(S_i, P_j)$: value of pheromone trail on edge from S_i to P_j

α : coefficient that controls importance level of pheromone

$\eta^{\alpha_l}(S_i, P_j)$: value of cost functions for parameter l for edge from S_i to P_j

α_l : coefficient that controls importance level of parameter l

The cost functions for all parameters for edge $S_i P_j$ are :

$$\eta^d(S_i, P_j) = \frac{1}{S_i P_j} \quad (13)$$

The equation (13) presents the cost function for distance $S_i P_j$ between the service station S_i and the parking P_j .

$$\eta^t(S_i, P_j) = \frac{1}{\text{tr}(D_i, S_j)} \quad (14)$$

The equation (14) presents the cost function for traffic where $\text{tr}(S_i, P_j)$ is the measured traffic flow from the service station S_i to the parking P_j .

$$\eta^f(S_{j_0}, P_k) = \text{sup}((\eta^f(D_i, S_{j_0}) - \gamma(D_i) * (S_{j_0} P_k)), 0) \quad (15)$$

S_{j_0} is the station that gives us the highest probability to go from D_i to a service station. $\eta^f(S_{j_0}, P_k)$ is the minimal fuel cost to reach the parking P_{j_0} .

$$\eta^v(S_i, P_j) = \text{Card}(P_j v) \quad (16)$$

where $P_j = P_{jv} \cup P_{j\tilde{v}}$ and $P_{jv} \cap P_{j\tilde{v}} = \emptyset$

P_{jv} is the set of available places in the parking P_j and $P_{j\tilde{v}}$ is the set of non available places in the parking P_j . $\text{Card}(P_j v)$ is the number of elements of the set P_{jv} .

This phase can be divided into two steps: a local and a global update [18]. The local update is done by each ant during the construction of the different routes. The global update of pheromone is performed when all ants reach parkings. The system will then be able to update all routes pheromone. The following rule (16) describes the process of the local update:

$$\Delta \tau_{ij}^k = (1 - \rho) \tau_{ij}^{\text{old}} + \tau_{ij} \quad (17)$$

$$\tau_{ij} = \sum_{k=1}^m \Delta \tau_{ij}^k = \begin{cases} \frac{Q}{L_k} & (\text{if ant } k \text{ use the edge } ij) \\ 0 & \text{else} \end{cases} \quad (18)$$

$(1 - \rho)$ is the pheromone reduction parameter ($0 < \rho < 1$). The parameter m is the number of ants, L_k is the length

of the tour performed by ant k and Q is a positive arbitrary constant set to be equal to 100.

V. DISTRIBUTED MULTI-OBJECTIVE ACO ALGORITHM APPLIED TO PARKING PROBLEM

In this article, a distributed system based on multi-agent is proposed to solve the parking problem using Multi-Objective ACO (Ant colony Optimization) algorithm. The main idea is to find an optimized path for the driver that will allow him to find a place in an indoor parking while respecting certain constraints like (Traffic ,distance ,amount of fuel in the car ,parking price , Parking Capacity).

A. Distributed architecture based on multi agent

We propose a distributed system to control the parking problem in the city, our solution is based on a set of intelligent agents which communicate and collaborate together in real time to provide the best solution. In Figure 3, we present our architecture to model the new distributed ACO algorithm for parking problem. In this solution, our system contains a set of intelligent agents, each one with a specific behavior. Different types of agents with different objectives have been used in this distributed architecture. The first type is the Ant Colony Optimization Agent (ACOA) which is divided into two agents:

-the first one is the Worker agent (WA) and its function is finding the optimal route to an indoor parking in the given map by respecting the constraints . The behavior of WA is defined by the set of visited nodes (current tour), the best tour encountered and the best tour for this agent to reach the parking.

-The second is the Controller agent (CA), its function is the search for optimizing the best solution. Each Controller Agent in the environment has some information like the coordinate of the driver, the coordinate of the parking, the coordinate of the service station, the distance, traffic between itself and each controller agent in the environment, amount of fuel in the car of the driver, the number of available places, parking cost. The pheromone quantity is stored in each Controller Agent, so each one knows the level of pheromone on each link to another agent.

-An other type is Master agent and its function is to collect informations from the Gui agent and send it to the master agent. This agent controls the communication between agents and send the current best tour to the Gui agent.

B. Multi-agent Interaction Model

The interaction model (figure 4) proposed consists of a series of messages such as:

The user initiates the map of the parking locations, driver location and service stations locations, The GUI Agent interact with the knowledge base to store all this information. Then send it to the Master Agent. The Master Agent receives the Map data and initial parameters .sends a request message to the ACO Agent which will apply the (ACO) Algorithm to search for the optimal route which fits preferences of the user.

The Worker Agent sends the origin node and current best path to controller agent to record the present path and calculates cost to each node. When it finished the CA sends the

list of neighbors, cost to each controller agent and best current path to worker agent. After that the worker agent chooses the next node and sends it to controller agent.

C. The Behavior of ACO Agent

In this solution, our system contains a set of intelligent agents; each one has a specific behavior. Three types of agents with different objectives were used in this distributed architecture. The first type is ant colony optimization agent (ACOA) and its function is constructing the new solution for finding the optimal routes to find a parking place in the given map by respecting the preferences of the driver. The behavior of ACO Agent is defined by The set of visited nodes (current tour), the best tour encountered and the best tour for this agent to reach the driver.

The second set of agents is the Manager agent (MA), its function is the search for optimizing the best solution. Each Manager Agent in the environment has some information like the coordinate of his parking, the number of places available in the parking, the number of reserved spaces, parking fees, the coordinate of other parking, also, the distance between itself and each Manager Agent in the city, and the distance between itself and the service stations in the city. The amount of pheromone is stored in each Manager Agent, so each Manager Agent knows the level of pheromone on each link to another agent.

The third type is Master agent and its function is to control the communication between agents and send the current best tour to the Interface agent

D. Ant colony algorithm based on multi agent system for solving parking problem

In this section, we present in detail the behavior of each intelligent agent in our distributed architecture. The Controller Agent has a collection of attributes: the quantity of pheromone, the level of evaporation, best solution cost, lists of nearest neighbor agents, the coordinate of the parking, the distance and traffic between itself and each controller agent in the environment, the number of available places, and parking cost. At each iteration, the Controller Agent records the present path. Table I show a pseudo-code of the Controller Agent behavior

TABLE I. THE CONTROLLER AGENT BEHAVIOR

INPUT: Graph, Initial parameters, Initial Preferences
OUTPUT : List of nearest nodes
<pre> GetAnt { RecordPathAnt(ant) CalculateCostFunctions(ant) UpdatePheromone(ant) ListBestneighbors(ant) Return BestNeighbors } </pre>

The behavior of CA is based on a set of functions

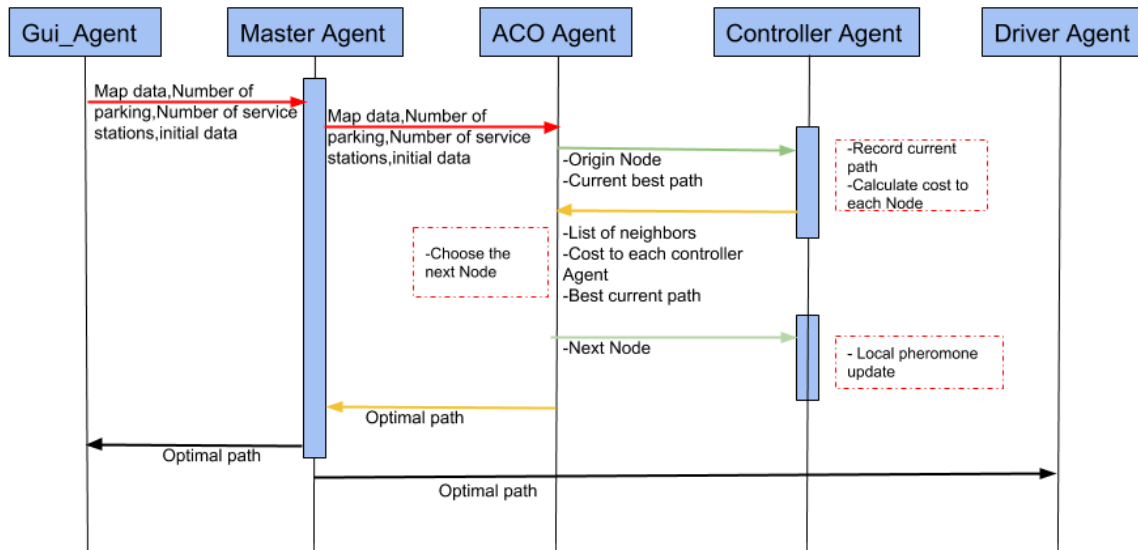


Fig. 4. Multi-agent Interaction Model

- The Controller agent saves the path of the present ACO Agent.
- Controller Agent calculates the cost functions for all problem parameters defined by the driver .updates the amount of pheromone for the received CA according to the information about its tour cost .
- The Manager agent constructs the list of best neighbors that will be sent to the Worker Agent for selecting the next node.

The objective of Worker Agent is computing the optimal path that connects a driver to an indoor parking in the city taking into consideration some constraints like (Traffic, distance, amount of fuel, Number of available places, parking cost). Table II presents the behavior of the Worker Agent.

- The Worker Agent(WA) Calculates the probabilistic transition to Select the next neighbor to visit and start a local pheromone update on this route. If the probabilistic transition of the path used by this agent is higher than the path stored at the current CA, the CA will update its path .
- At the end, the WA compares his solution with the best solution in the environment .If the cost of the path used by the WA is less than the best path in the environment, the WA will update the global amount of pheromone on this route and this path will be the best solution in the environment.

VI. FUTURE WORK

In the future work we will implement our solution using Java-Agent Development Environment (JADE) as a distributed and parallel platform. The proposed solution is deployed on several containers. At the beginning, we will need to fix

TABLE II. THE BEHAVIOR OF THE WORKER AGENT

INPUT:Graph,Initial parameters
OUTPUT : Optimal path
<ul style="list-style-type: none"> • OptimalPath() • UpdateGlobalPheromone() • DisplayBestPath() <pre> OptimalPath() { CalculateProbabilisticTransition() SelectBestTransition() UpdateLocalPheromone() UpdateListBestPath() Return Bestpath }</pre>

the number of iterations. Each iteration ends when a set of Ant agents reaches an indoor parking and the parameters are initialized at the start of the iteration. In each iteration we collect the data (the best cost, the amount of pheromone, the best solution ...). At the end of each round, we can use these data to find the optimal solution for the driver taking into consideration network constraints like (Traffic, distance, amount of fuel, cost of parking, number of available places in the parking).

VII. CONCLUSION

This paper presents a distributed solution based on multi objective ACO and multi agent systems to solve multi constraints parking problem .The proposed solution uses the behavior of real ants based on artificial agents .

The new solution was applied to solve the parking problem

by creating an optimal that feeds the preferences of the driver (Traffic, distance, amount of fuel, Number of available places in the parking, parking cost). The Parking problem is presented as a graph with multiple parking, each one is managed by a Manager agent and the process of communication between agents is described. In our proposed architecture, agents work together in parallel to find the optimal path for the driver. When one of the agents fails in his behavior, the system keeps running because we have a set of agents available to achieve this task. as future works we will consider other forms of parallel Ant Colony optimization algorithm and we will use this algorithm to solve other problems.

ACKNOWLEDGMENT

I would like to thank my advisors Ms.W.Dachry, Mr. F. Moutaouakkil and Mr. H. Medromi for the efforts they made to review this paper and for the many useful suggestions they gave me during the work on this paper.

I would also express my gratitude to every person that has contributed to the achievement of this works.

REFERENCES

- [1] Y. R. Rao, "Automatic smart parking system using internet of things (iot)," *Int J Eng Technol Sci Res*, vol. 4, no. 5, 2017.
- [2] Z. Suryady, G. R. Sinniah, S. Haseeb, M. T. Siddique, and M. F. M. Ezani, "Rapid development of smart parking system with cloud-based platforms," in *Information and Communication Technology for The Muslim World (ICT4M), 2014 The 5th International Conference on*. IEEE, 2014, pp. 1–6.
- [3] B. M. K. Gandhi and M. K. Rao, "A prototype for iot based car parking management system for smart cities," *Indian Journal of Science and Technology*, vol. 9, no. 17, 2016.
- [4] I. Samaras, A. Arvanitopoulos, N. Evangeliou, J. Gialelis, and S. Koubias, "A fuzzy rule-based and energy-efficient method for estimating the free size of parking places in smart cities by using wireless sensor networks," in *Emerging Technology and Factory Automation (ETFA), 2014 IEEE*. IEEE, 2014, pp. 1–8.
- [5] Y. Zheng, S. Rajasegarar, and C. Leckie, "Parking availability prediction for sensor-enabled car parks in smart cities," in *Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on*. IEEE, 2015, pp. 1–6.
- [6] Z. Ali and W. Shahzad, "Comparative analysis and survey of ant colony optimization based rule miners," *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 1, 2017. [Online]. Available: <http://dx.doi.org/10.14569/IJACSA.2017.080108>
- [7] H. Salehinejad, F. Pouladi, and S. Talebi, "A new route selection system: multiparameter ant algorithm based vehicle navigation approach," in *Computational Intelligence for Modelling Control & Automation, 2008 International Conference on*. IEEE, 2008, pp. 1089–1094.
- [8] H. Salehinejad and S. Talebi, "A new ant algorithm based vehicle navigation system: a wireless networking approach," in *Telecommunications, 2008. IST 2008. International Symposium on*. IEEE, 2008, pp. 36–41.
- [9] J. Yang, J. Portilla, and T. Riesgo, "Smart parking service based on wireless sensor networks," in *IECON 2012-38th Annual Conference on IEEE Industrial Electronics Society*. IEEE, 2012, pp. 6029–6034.
- [10] H. C. Yee and Y. Rahayu, "Monitoring parking space availability via zigbee technology," *International Journal of Future Computer and Communication*, vol. 3, no. 6, p. 377, 2014.
- [11] O. Orrie, B. Silva, and G. Hancke, "A wireless smart parking system," in *Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE*. IEEE, 2015, pp. 004 110–004 114.
- [12] E. Karbab, D. Djenouri, S. Boulkaboul, and A. Bagula, "Car park management with networked wireless sensors and active rfid," in *Electro/Information Technology (EIT), 2015 IEEE International Conference on*. IEEE, 2015, pp. 373–378.
- [13] J.-H. Shin and H.-B. Jun, "A study on smart parking guidance algorithm," *Transportation Research Part C: Emerging Technologies*, vol. 44, pp. 299–317, 2014.
- [14] H. Wang and W. He, "A reservation-based smart parking system," in *Computer Communications Workshops (INFOCOM WKSHPS), 2011 IEEE Conference on*. IEEE, 2011, pp. 690–695.
- [15] W. Cai, D. Zhang, and Y. Pan, "Implementation of smart parking guidance system based on parking lots sensors networks," in *Communication Technology (ICCT), 2015 IEEE 16th International Conference on*. IEEE, 2015, pp. 419–424.
- [16] N. Mejri, M. Ayari, R. Langar, and L. Saidane, "Reservation-based multi-objective smart parking approach for smart cities," in *Smart Cities Conference (ISC2), 2016 IEEE International*. IEEE, 2016, pp. 1–6.
- [17] M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," *Theoretical computer science*, vol. 344, no. 2-3, pp. 243–278, 2005.
- [18] M. Dorigo and L. M. Gambardella, "Ant colony system: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on evolutionary computation*, vol. 1, no. 1, pp. 53–66, 1997.