Ant Colony Optimization for an Adaptive Transportation System: A New Termination Condition Definition Using an Environment Based Approach

1st Pierre Romet
CIAD (UMR 7533)
UTBM
Belfort, France
pierre.romet@utbm.fr

2nd Franck Gechter
CIAD (UMR 7533) and MOSEL LORIA UMR CNRS 7503
UTBM and Universite de Lorrain
Belfort and Vandoeuvre, France,
franck.gechter@utbm.fr

3rd El-Hassane Aglzim
ISAT-DRIVE
Univ. Bourgogne Franche-Comte
Nevers, France
el-hassane.aglzim@u-bourgogne.fr

4th Olivier Grunder
NIT-O2S
Univ. Bourgogne Franche-Comte
Belfort, France
olivier.grunder@utbm.fr

5th Sidi-Mohammed Senouci
ISAT-DRIVE
Univ. Bourgogne Franche-Comte
Nevers, France
sidi-mohammed.senouci@u-bourgogne.fr

***Abstract*—The delivery of the packages following the online purchase of goods trough web giants platforms is growing faster each years. To meet the demand of the growing quantity of packages and their delivery, the algorithm used to resolve the Vehicle Routing Problem (VRP) has to be efficient and adaptive. The algorithms used to solve the VRP algorithm still provide better turn, but do not deal with situation adaptation at delivery point. Seeking to fit into this adaptive feature, the commitment of this paper is to lay solid groundwork for the development of an adaptive transportation system. Exploring various strategies taking care of the possibilities of delivery at delivery point, our objective is to define a strategy minimizing the time and the distance travelled by packages and maximizing the satisfaction of the customers.**

***Index Terms*—Ant Colony convergence; Adaptive delivery**

I. INTRODUCTION

The research presented in this paper is part of the development of a new urban and rural freight transportation system. Each year, the amount of packages bought on web market places grows, to the detriment of local urban and rural shopkeepers. Our freight transportation system aims to revitalize urban and rural areas by providing to the customer the possibility to buy online goods from local shops and restaurants and to benefit from delivering services similar to those proposed by traditional web market places. Moreover, our project is intended to be more than a classical transportation system. Seeking to revitalize the urban area and to draw the attention of as many customers as possible, the delivery of goods has to be as close as possible to customers in city center and rural area. The project try to provide adaptive feature to the delivery services, to consider the customer as the most important part of the system.This delivery transportation system is described in detail in [3]. The main principle of this system is based on small electrical vehicles able to follow the buses or the tramways of the already existing public transportation system using platoon control algorithm. As soon as one of these vehicles arrives to a stop near a delivery target, it can park near the bus stop and then wait for the customer. The main interest of this system is to have a potential ability to adapt to the load (the number of goods to be delivered) and to the dynamical constraints of the customers. Thus, to implement this kind of freight transportation system, we will need to adopt and implement applications of Vehicle Routing Problem. First of all, we need to define a good base structure on which to implement our transportation system. Our choice focused on the Ant Colony to performed on the VRP algorithm, thanks to its intrinsic ability of adaptation to the dynamic update of the graph. This ability provides us the capacity to include real time adaptation to the demand of delivery and pickup of packages. However, wishing to develop an adaptive freight transportation system, the ant colony algorithm needs to be self-adaptive convergent. In this context, the ant colony algorithm defined in [1] does not fit directly in this self-adaptive convergent feature. Thus, the first part of our work is to identify if the self-convergence of the ant colony algorithm can be characterized by an environment based approach. This research work is developed in the first Section II of this paper. Then, after having defined the self-adaptive convergent property of the ant colony algorithm, the next step is to focus on the implementation of the VRP application to our freight transportation system. Basing our application on the VRP, a way to bring an adaptive feature to it is to find one. To do so, we include in the VRP application different strategies to

 ISBN: 978-1-61208-781-8

adapt the possibility of delivery at city point according to the adaptive possibility of customers to be present to pickup the package(s). Thus, those strategies are not limited to deliver or not a package(s) according to the presence of the customer, it must provide the customer the possibility to redefine his point of delivery and its time of delivery during the same turn of the vehicle. According to this fact, the delivery turn, defined by the VRP application has to self-adapt its path to deliver package(s). This research work is defined in Section III of this paper.

The paper is structured as follows: Section II provides the overview of the basics of ant colony optimization and concludes with the environment based approach to define the self-adaptive convergent property of the algorithm. Section III gives an overview of the Vehicle Routing Problem approach and develops self-adaptive reconfiguration to adapt its turn to the availability of the customer. Finally, we will present our experimentation, results and analyses in Section IV and then Section V concludes the research conducted in this paper.

II. Ant Colony Algorithm Convergence

This section provides an overview of the Ant colony Algorithm before introducing the work on the convergence criterion of this algorithm.

A. Ant colony for operational research

This section is intended as a review of the main features of the ant colony optimization algorithm, that Dorigo synthesized in [6]. To do this, we will support our demonstration by the determination of the shortest path between two points; a combinatory optimization problem already widely covered in the specific literature. For a set of "n" cities, a shorter path search allows to determine the smallest distance between two cities, passing through each of them once. The shorter path search is based on the implementation of a graph $G = (N, L)$, with N all cities and L all paths connecting cities. Each arc $li \in L$ having a d_{ij} value that characterizes the distance between two cities i and j. Optimization by ant colony is the study of how work guides the worker [4]. Ant agents use pheromones that guide them on the paths with the shortest distance between the anthill and the food source. This solution is built by a succession of turns in which an agent travels guided by a pheromone trace and a search heuristic. Then, when all ant agents have finished their turn, they come back to the anthill by updating their pheromone trace. Ant agents thus seek a better solution within the paths that have the highest pheromone rate.

1) Iteration construction: When the algorithm is initialized, all ants (the number of ants being equal to the number of cities in the [5] graph) are dropped to the common starting point. Then, each ant agent applies a stochastic search, called the "random choice rule," to determine which to which city it will move on thereafter. Taking k agent ants, in a city i, the next city will be chosen with the probability defined below.

$$P_{ij}^k(t) = \begin{cases} \frac{[\tau_{ij}(t)]^\alpha.[\eta_{ij}(t)]^\beta}{\sum\limits_{r \in N_j^k}[\tau_{ir}(t)]^\alpha.[\eta_{ir}(t)]^\beta} & if \quad j \in N_i^k \\ 0 & otherwise \end{cases} \tag{1}$$

With $n_{ij} = \frac{1}{d_{ij}}$ a heuristic value, $T_{ij}(t)$ the value of the pheromone trace, with t the iteration counter. Side and Side parameters determine the influence of pheromone trace and the heuristic value. Finally N_{ik} represents the direct neighborhood of the ant agent k, being in city i.

2) Pheromones update : Once all the ants have completed an exploration, they must return to their starting town and will at the same time update the pheromones rate of the paths forming their route during the last round. The pheromones are updated using the following equation:

$$\tau_{ij}(t+1) = [(1-\rho) * \tau_{ij}(t) + \sum_{k=1}^{m} \Delta\tau_{ij}^k(t)] \tag{2}$$

With $p(0 < p <= 1)$ evaporation rate, $\Delta\tau_{ij}^k$ the amount of pheromones that the agent k repents on the path visited, as follows:

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{1}{f(S^k)} \text{if the edge belong to } S^k \\ 0 \text{ otherwise} \end{cases} \tag{3}$$

where $f(S^k)$ is the size of the path constructed by the ant k agent, for the current iteration.

TABLE I. PROCEDURE ACO ALGORITHM

Procedure ACO algorithm
set parameters, initialize pheromone trails
while (termination condition not met) do
construct ants solutions
update pheromones
end
end

The pheromones act as a strengthening position, encouraging other ants to use an already marked path, amplifying the positive reinforcement effect as more and more ants go along the path.

3) Termination condition: Since the development of optimization algorithms by ant colony, the efforts made through various research projects have focused on the development of iteration construction and pheromone updating. However, very few have sought to develop new approaches on how to end algorithms. In addition, it is legitimate to ask which methods to put in place, since the number of optimal solutions available is infinite.

B. State of the art of termination criteria

By the stochastic property of ant colony algorithms, the emergence of a solution for a given problem is guaranteed, with a probability of reaching 1, for an infinite time of exploration. Such application is not possible, therefore it is necessary to agree on the number of iterations to end ACO, to enable the emergence of an optimal result that best characterizes the exploited problem. The number of iterations being the key, the higher the iteration, the better the quality of the solution. However, arbitrarily setting a very high number of iterations is very inefficient in terms of resources, whether they are energy and/or computational. Therefore, it appears to be possible to use as little iteration as possible to determine the best solution. In addition, this search by means economy is adequate within this meta-heuristics, as illustrated by [7] through the "Stalling effect", which they define as the following: "the problem of stalling effect in fitness function is related to the non improvement of the fitness value during the evolutionary process". Since this phenomenon has been observed through many algorithm run, it appears that a better solution emerges at a given point in the research. However, the latter tends to stagnate before the simulation stops, without a better solution emerging again.

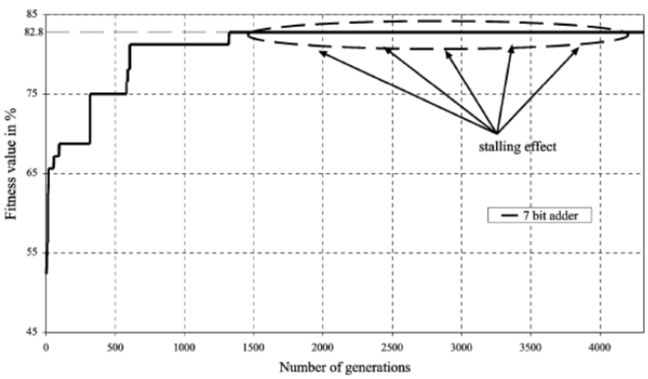

Fig. 1: Stalling effect

The optimization of the resources of ACO is entirely based on the way in which the termination criteria will be set. There are various methods in the scientific literature that we will now explain. Currently one of the most common approaches is to use an arbitrarily set termination criteria. This results in a high number of iterations, but above all,they are defined randomly. When the specified number of iterations is reached by the algorithm, the algorithm will return the best solution found [13]. Another, but similar, approach is to stop the iteration of the algorithm after it has exceeded a pre-defined execution time, always randomly [10]. In the end, these two approaches, although easy to put in place, reveal a complete decorrelation between the agents (ants) and their environment, cutting them off from the influence of the latter. In addition, they translate in a highly resource-intensive implementation, which is antinomic with the principle of optimization sought through meta-heuristics. Statistical approaches using evolutionary factors, are then used to evaluate the termination criteria of the algorithm to terminate its execution. Nicolas et al. [9] have developed an iterative solution that involves launching their ant colony algorithm several times conditioned by a minimum number of iterations to reach. Then they determine a percentage of success by comparing the different solutions obtained according to the parameters of each execution. Zhaojun et al. [14] propose a solution characterizing the evolution of the system using three factors. The first two assess whether the termination criteria is met and the last one is also an assessment factor based on the convergence of path with pheromone around the shortest path. In addition to the above methods based on the use of an arbitrarily fixed termination criteria, solutions based on decision trees were also developed. Kate et al. [2] have developed a decision tree that determines the best heuristics to be implemented to solve the problem facing the latter. Kate et al. [12], have also worked on the ontology engine allowing to choose the best stop condition, allowing to optimize the termination criteria. Moreover, by exploring the literature to find advanced methodologies for our problem, a physics-centered solution was found. This solution proposes to implement the mechanics of the point through the observation of position curves, speed and acceleration. The study of these curves is supposed to help in the decision-making process to stop the execution of the algorithm [8]. To finish this review of various approaches available in the literature, we will now discuss the search for the local minimum. The solution provided by [11], proposes to stop this search for a solution when a better solution is found, coupled with an arbitrary stop criteria in case the algorithm does not converge. However, nothing is defined as to how a better solution is evaluated, or even how it is correlated with the environment in which the research is conducted. Finally, the approach that we propose in this article is also based on this idea of local minimum exploration, like the solution of Silvia et al. [11]. However, unlike the latter, we strive to correlate the depth of exploration of the local minimum with the environment in which our agents operate. By doing so, we seek dynamic behavior for the algorithm, giving it an adaptation to its environment to help it make decisions about its termination criteria. By doing so, we are trying to demonstrate that a good knowledge of the studied environment can bring real added value in the search for solutions.

C. Environment based approach for the ant colony convergence

The objective of this section is to provide a self adaptive convergence to the Ant Colony algorithm. As seen previously, there is many different approach used to stop the ACO, but most of those amount to end up with a termination criteria arbitrarily set, to limit the solution. Wishing to provide an adaptive feature to the Ant Colony algorithm, the question was: "How to determine an adaptive termination condition and on what it can be based on?". Each project on which an ACO can be applied is different from an other, but they all have pieces in common, the structure of their graph. The uniqueness of each graph can be reflected through its number of vertices, its number of edges or even its complexity. Thanks to the last research, we could demonstrate that the number of edges ant the number of vertices do not characterize the convergence of the Ant Colony algorithm. Finally, the only lead we left to study for an adaptive termination condition is the complexity of the graph. To conduct this study, we implement two different benchmark tests on which to base our research. The first one is a small graph, made of 10 vertices, but with a significant complexity compared to its small number of vertices. The second is a reproduction of the public transportation system of the city of Belfort. This graph, although very voluminous, with 150 vertices, has a low complexity. Thanks to those graphs, we conducted experiments that generated raw data to exploit. In the last section IV we will see if the generated raw data allow us to correlate graph complexity and adaptive Ant Colony termination.

III. ADAPTIVE TRANSPORTATION SYSTEM AT DELIVERY POINT

This section provides an overview of the Vehicle Routing problem and introduce different strategies to provide an adaptive feature to this classical problem.

A. Vehicle Routing Problem overview

As introduced previously, our commitment trough this paper is to provide adaptive delivery feature to the Vehicle Routing Problem. But first of all, we will define the Vehicle Routing Problem application. Thank to the previous section II-A, we already describe the Ant Colony part of the VRP application such as choice of a new point to reach, and update of the pheromone trails. However, the VRP obeys to additional rules that we will introduce now. For each customer v_i, a non-negative demand q_i is given ($v_i = 0$). The aim is to find a minimum cost vehicle routes where:

- Every customer is visited exactly once by exactly one vehicle
- All vehicle routes begins and end at the same depot
- For every vehicle route, the total demand does not exceed the vehicle capacity Q

The VRP is a very complicated combinatorial optimization problem that has been studies since the late fifties because of its central meaning in distribution management. Problem specific methods as well as meta-heuristics like tabu search, simulated annealing, have been proposed to solve the VRP. VRP and TSP are closely related. As soon as the customers of the VRP are assigned to vehicle, the VRP is reduced to several TSPs. For that reason, our approach is highly influenced by the TSP ant system algorithm by Dorigo [1]. To solve the VRP, the artificial ants construct vehicle routes by successively chosing cities to visit, until each city has been visited. Whenever the choice of another city would lead to an infeasible solution for reason of vehicle capacity or an already visited point, the depot is chosen and a new VRP turn is started. Concerning the initial placement of the artificial ants, it was found that the number of ants have to be equals to the number of vertices [5] (for the TSP and VRP) and each ant should start its turn at different vertices of the graph.

B. Strategies for adaptive delivery

As defined previously, our application is based on the VRP, but with the need to providing it adaptive feature. To do so, we paired the VRP application with different strategies to adapt the possibility of delivery, at city point, according to the adaptive possibility of customers to be presented to pickup his package(s). Thus, those strategies are not limited to deliver or not a package(s) according to the presence of the customer, it has to provide to the customer the possibility to redefine its delivery time at its delivery point during the same turn of the vehicle. According to this fact, the vehicle processing the delivery turn, define by the VRP application, has to self-adapt the scheduling of his set list of city points to deliver package(s).

1) First Strategy: The first strategy is a classical VRP turn to have a point of reference to compare the results with those of the second strategy. This strategy is common to all delivery city points, is defined as: ”deliver the package(s) and continue”.

2) Second Strategy: The second strategy is defined as follow:

- Upon arrival at delivery point, the vehicle has two choices:
  - If the customer is present, the vehicle delivers the package(s) and continue
  - If the customer is not present, the vehicle waits 10 minutes
    - If the customer arrives, the vehicle delivers the package(s) and continues
    - If the customer do not arrive, the vehicle continues and has to come back at the end of its VRP turn.

Furthermore, we implement two variants for this strategy. At the end of a turn, if all the packages were not delivered we organize the second part of the turn to be sure to deliver them.

First variant:

- The current VRP turn is ending at the depot (its starting point)
  - A possibility is create a new turn thank to the VRP algorithm
  - A other one is to link all points with a shortest path algorithm

Second Variant:

- The current VRP turn stop at the last delivery point of its turn
  - A possibility is create a new turn thank to the VRP algorithm
  - A other one is to link all points with a shortest path algorithm

3) Evaluation Criteria: For this work, the evaluations criteria will be the following:

- The total time:
  - of the VRP turn
  - for a delivery
- The total distance:
  - of the VRP turn
  - for a delivery

 ISBN: 978-1-61208-781-8

IV. Experiments and Results

As presented at the beginning of this paper, the development of our transportation system is based on an adaptive Vehicle Routing Problem to self adapt to the possibility of delivery of goods at city points. Our transportation system being based on an Ant Colony to support VRP algorithm, the first stage is to improve the Ant Colony Algorithm to bring it an adaptive convergence. This point will be developped following this paragraph, in IV-A. Next, to bring a self-adaptation to the possibility of delivery at city point, we have to develop an adaptive VRP in order to answer our needs of improvement for our transportation system. This part will be analyzed and develop in the section IV-B.

A. Environment based approach for the ant colony convergence

In a previous work, we explored two different paths to characterize the convergence of the Ant Colony algorithm. Searching to exploit an environment based approach, the objective was to identify if the number of vertices or the number of edges could be the key for the self convergence of the Ant Colony Algorithm. However, as we demonstrate it, those parameters were not the key. In this section, the objective is to identify if the last parameter of an environment based approach, being the graph complexity, can solve the self convergence of the Ant Colony Algorithm.

We defined the graph complexity as the average number of edges connecting a vertex to others in the graph. As said previously, our test bench is based on two graph, the first being a small graph, but with a significant complexity. The second one, being a reproduction of the public system transport of the city of Belfort, with a low complexity.

The first step was to define the complexity of any of our graph. To do that, we calculate the complexity of our two graphs, calculating the following value:

$$\theta = \frac{\sum_{v=1}^{n} \omega_i}{k} \tag{4}$$

with v the number of vertices into the graph, ω_i the number of edges starting to the current vertex and connecting it to an other vertex and k the number of edges into the graph.

Thus, thanks to θ, we determined that $G_1(v, e)$ has a complexity of 1.25, and $G_2(v, e)$ has a complexity of 1.05. Having been able to determine the complexity of any of our graph, we will now search a way to define the number of iteration ϵ of the Ant Colony, according to the graph complexity previously calculated. So, to identify ϵ, we took inspiration from work we previously performed in which we tried to identify if the number of algorithm iterations can be reflected by the number of ants into the graph. Starting with one ant and increasing the number of ants until it reach the number of vertices (according to [5], the number of ants should be equals to the number of vertices in a graph).

Those results gave us the followings graphics:

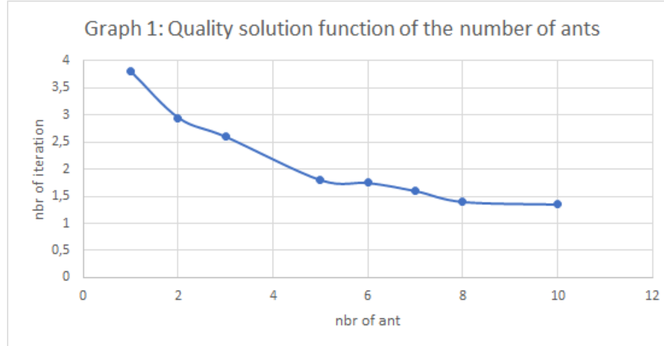

Fig. 2: Quality solution of the test graph

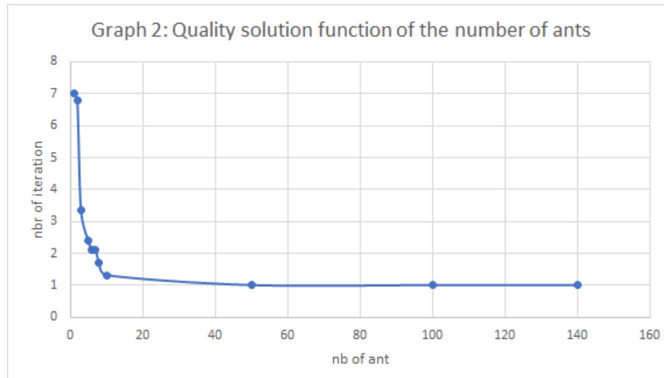

Fig. 3: Quality solution of the graph of Belfort

The "Quality solution" illustrate the number of algorithm iterations according to the number of ants. For the tow previous simulations, the number of iteration was arbitrarily set to a high value. This allow us to identify the stagnation point of the stalling effect curve and determine the best number of iterations for the explored problem. Thanks to previous work, we refuted the claim saying that the convergence of the ACO can be defined by a number of ants equals to the number of vertices. As it can be observed in Figure 4 and 5, the two curves function of our two graphs do not have the same equation. However, as it can be observed in Figure 2 and 3, the graph complexity is the key. Considering the graph complexity as the optimisation point to reach, paired with the Dorigo criterion [5], we obtain the convergent point for our two curves which characterizes the self-convergence of the Ant Colony algorithm.

B. Adaptive transportation system at delivery point

As introduce it earlier, simulations was leads on two strategies. The first one being a classical VRP application allowing to find the best turn according to the set list of city points, paired to the obligation of delivery at each city point. This first simulation gave us a reference total distance travelled and a reference total time for the vehicle turn. Those distance and time references are used as benchmark to analyse the performance of the second strategy. The second strategy being composed of two variants to explore different possibilities

 ISBN: 978-1-61208-781-8

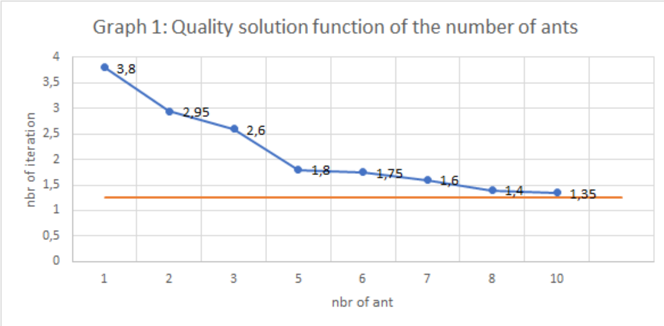

Fig. 4: Quality solution test graph compare to complexity solution

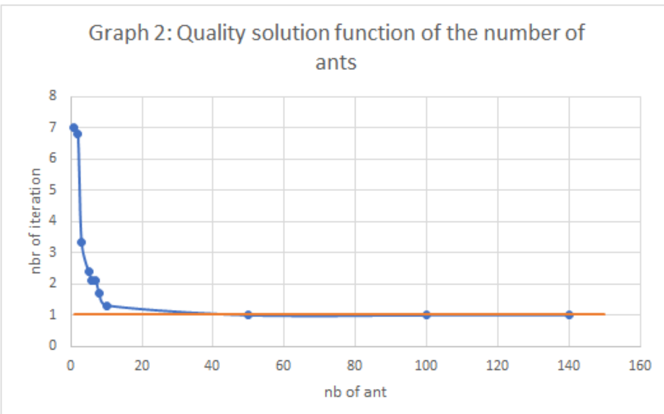

Fig. 5: Quality solution Belfort graph compare to complexity solution

through an adaptive VRP algorithm. Simulations was lead on $G_1(v, e)$, (previously defined in this paper) with one package delivered by vertex.

TABLE II. Procedure ACO algorithm

Strategies	total distance (km)	total time (min)
Strategy 1	44	29
Strat2: frst variant		
cascading VRP	52	114
VRP and ShP	49	110
Strat2: scd variant		
cascading VRP	44	102
VRP and ShP	41	98
VRP = Vehicle Routing Problem, ShP = Shorstest Path		

According to Table 2, the first strategy provides the best indicators in terms of distance, with a travel distance of 44 km and time, with a turn time of 29 min. However, the classical VRP is not adaptive, so it does not take into account the un-delivery possibility of package(s). Thus, at the end of a turn, the total percentage of un-delivered package(s) will necessary impact the next turn, by overloading the delivery delivery turn and/or postpone the delivery of future package(s). So, wishing to perform an adaptive application, the objective of the second strategy is to be able to ensure the delivery of the totality of packages for a given turn. The second strategy explores two variants for the "re-starting" point of the un-delivered package(s) turn. The first variant is to come back to the goods deposit. After that, the last turn is define either with a reload of the VRP algorithm of with a shortest path, linking all un-delivery city points. Regrettably, this variant violently deteriorates distance and time indicators. The second variant offers to start the un-delivered package(s) turn at the last city point visited by the vehicle. After that, the last turn is define either with a reload of the VRP algorithm of with a shortest path, linking all un-delivery city points. As can be seen in Table 2, this variant clearly proves his interest. Although this solution deteriorates the time indicators (reflecting a longer use of the vehicle), the distance indicator stays stable in the worst case (44 km) and is even improved, with a gain of 7%, while paired with the use of shortest path algorithm, all that with a delivery rate of 100%.

V. CONCLUSIONS

Following the re-contextualization of the ant colony algorithm in the section II-A, we provide a non-exhaustive state of the art regarding the criterion termination of the ant colony algorithm in the section II-B. Then, we search to know if an environment based approach can characterize the self-adaptive convergence of the ant colony. In previous work, we concluded that the others parameter of the graph for an environment based approach was not conclusive. Thus, into the section II-C, thanks to the study, we could demonstrate that the graph complexity was the good way to illustrate self-adaptive convergence of the algorithm. Then, with the section III we explored different strategies to provide self-adaptive ability for the VRP turn, to be more flexible and to match as well as possible to the availability of customers to pick-up their package(s). Finally, into the section IV we present our results for the self-convergence of the ant colony algorithm in section IV-A and for the adaptive delivery VRP turn in the section IV-B. Furthermore, even if the time indicator is deteriorated, the improvement of the distance indicator support the use of electric vehicle for future research. The next step will be to combine the Vehicle Routing Problem with the capacity of the vehicle to characterize VRP turns and to define feasible adaptation during VRP turn.

VI. ACKNOWLEDGMENT

This work is done with the support of the Région Bourgogne-Franche-comté, through the SURATRAM Project.

REFERENCES

[1] V. Maniezzo A. Colorni, M. Dorigo. Distributed optimization by ant colonies. In *Proceedings of the First European Conference on Artificial Life*, 01 1991.

[2] J. Giffin Y. Tu A. Kate, R. James. A knowledge discovery approach to understanding relationships between scheduling problem structure and heuristic performance. In *Learning and Intelligent Optimization*, 2009.

[3] S. Senouci N. Kroichvili C. Cappelle D. Fass F. Gechter, E. Aglzim. Transportation of goods in inner-city centers: Can autonomous vehicles in platoon be a suitable solution? In *2017 IEEE Vehicle Power and Propulsion Conference (VPPC)*, pages 1–5, 2017.

[4] P. Grassé. La reconstruction du nid et les coordinations interindividuelles chezbellicositermes natalensis etcubitermes sp. la théorie de la stigmergie: Essai d'interprétation du comportement des termites constructeurs. *Insectes Sociaux*, 6:41–80, 2005.

[5] A. Colorni M. Dorigo, V. Maniezzo. Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26(1):29–41, 1996.

[6] T.Stützle M. Dorigo. Ant colony optimization algorithms for the traveling salesman problem. *Ant Colony Optimization*, pages 65–119, 2004.

[7] P. Broomhead M. Veluscek, T. Kalganova. Improving ant colony optimization performance through prediction of best termination condition. In *2015 IEEE International Conference on Industrial Technology (ICIT)*, pages 2394–2402, 2015.

[8] P. Broomhead M. Veluscek, T. Kalganova. Improving ant colony optimization performance through prediction of best termination condition. In *2015 IEEE International Conference on Industrial Technology (ICIT)*, pages 2394–2402, 2015.

[9] I. Stéphan P. Nicolas, F. Saubion. Answer set programming by ant colony optimization. In *Logics in Artificial Intelligence*, 2002.

[10] A. Rizzoli A. Donati R. Montemanni, L. Gambardella. Ant colony system for a dynamic vehicle routing problem. *Journal of Combinatorial Optimization*, 10:327 – 343, 2005.

[11] I.Loiseau S. Mazzeo. An ant colony algorithm for the capacitated vehicle routing. *Electronic Notes in Discrete Mathematics*, 18:181 – 186, 2004.

[12] X. Lim S. Miles, J. Hemert. Understanding tsp difficulty by learning from evolved instances. In *Learning and Intelligent Optimization*, 2010.

[13] P. Abad Y. Gajpal. An ant colony system (acs) for vehicle routing problem with simultaneous delivery and pickup. *Computers Operations Research*, 36(12):3215 – 3223, 2009.

[14] Z. Ren Z.Zhang, Z. Feng. Approximate termination condition analysis for ant colony optimization algorithm. In *2010 8th World Congress on Intelligent Control and Automation*, pages 3211–3215, 2010.

ISBN: 978-1-61208-781-8