International Journal of Computer Science and Applications

Vol. 10, No. 2, pp. 1 – 14, 2013

ASSIGNMENT OF CONTAINER TRUCKS OF A ROAD TRANSPORT COMPANY WITH CONSIDERATION OF THE LOAD BALANCING PROBLEM

GHIZLANE HAMDOUNI

University of Hassan II
Casablanca, Morocco
Hamdouni.ghizlane@gmail.com

AMINA ELOMRI

University of Hassan II
Casablanca, Morocco
a.elomri@fsac.ac.ma

KHALID MOUSSAID

University of Hassan II
Casablanca, Morocco
k.moussaid@fsac.ac.ma

In a container trucking company, the construction of a schedule of reservations is to assign to each reservation, optimally, one or more container while meeting the deadlines proposed by the customers, the employment regulations and the availability of trucks.

In this paper, we deal this problem of assignment which consists of minimizing the total distance traveled by trucks while satisfying all customer demands and taking into account the load balancing problem which is a crucial constraint because some vehicles can consume their potential more or less faster than the rest of the trucks and the company can find itself with a significant number of trucks unable to work.

After modeling of the problem posed by our trucking company of medium size (about 400 trucks), we propose to solve it with an exact method compared with a heuristic one based on ant colony algorithm. The decision-making tool proposed will allow the logistic manager a better management of the truck fleet. Several tests, on real data from a trucking company International medium, show the efficiency of the proposed method.

Keywords: Road Transport; Logistics; Truck; Assignment; Modeling; Optimization; ant colony algorithm.

1. Introduction

The trucking companies are facing a lot of requirements to survive in a market where competition is very strong, that forcing them to reduce the direct and indirect costs and to increase quality in order to better meet customer needs.

We are interested in this article in the problem of assignment of container trucks to different customer bookings, while respecting the deadlines proposed by the customers, and responding to all requests made (§ Fig. 1).

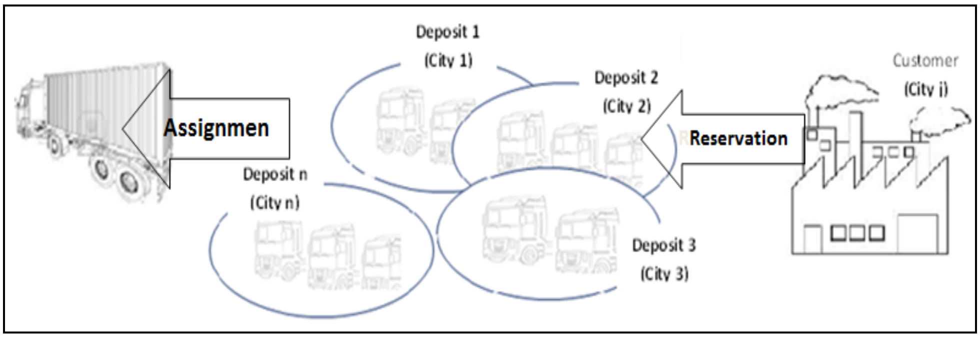

Fig 1. Assignment steps of the container trucks.

N.B. we suppose that during a reservation, there will be always a sufficient number of available trucks which can satisfy the demands of the customers.

Our problem is to determine a set of tours that minimize the variable operational costs and that can be classified in our case in two categories: the raw material "fuel" and the cost of unplanned aging of vehicles (the cost engendered by the unbalanced of load of trucks).

In the literature, as any combinatorial optimization problem, the problem of assignment of vehicles has been studied and solved by exact methods and heuristics or meta-heuristics.

[Rego(1994)] and [Iori et al. (2007)] studied the problem of assignment of vehicles with an exact method using the Branch & Bound algorithm which is effective only on the problems having less than 25 customers (for [Rego(1994)]) and 30 customers (for [Iori et al. (2007)]).

[Grellier(2008)] propose an exact method based on the linear programming, and summarizes the total cost of routing in two costs: the fixed cost of vehicle usage and the kilometric cost-in-use, however he didn’t take into account the factor of load balancing between the vehicles.

[Khebbache et al. (2010)] proposed memetic algorithm which is a genetic algorithm hybridized with a local search to minimize the total distance traveled. His solution improves the results of heuristic on all instances, with a percentage of overall improvement of 21.32 %, but , but without taking into account the cost of load balancing.

[Dorigo et al. (2004)], [Solnon (2010)] and [Lacomme et al. (2010)] and [Khichane (2009)] dealt with the assignment of vehicles with ant colony algorithm, but without taking into account either the cost of load balancing.

These different methods and studies were interested in the problem of assignment of vehicle whose objective is to minimize the total distance traveled. Another objective for trucking companies that have been treated by none of these authors is the load balancing of travel trucks. This objective is indeed crucial because some vehicles can consume their potential more or less faster than the rest of the trucks and the fleet will not remain homogeneous. In this case the company can find itself with a significant number of trucks unable to work.

In the present article, we take into account in our assignment method this problem of load balancing.

After analysis and modeling of the problem, we propose to solve our problem of assignment of trucks with an exact method " ATEM " and a heuristic method based on the principle of ant colony "ATAC", then we propose a decision-making tool, which allows to choose the truck to be affected while respecting the constraints of the problem, minimizing the total distance traveled by trucks and taking into account the balance of their loads.

The various simulations, with real data given by the road transport, illustrate the effectiveness of the proposed methods and show that our algorithms provide a good compromise between computation time and quality of solutions generated.

2. Modeling of the Problem

After studying the existing, we defined the variables of the problem and modeled the constraints to respect and the objective to satisfy. So, we propose for the problem of the trucking company, a model based on the linear programming with Boolean variables X_{ij}. The Variable X_{ij} is worth 1 if the reservation i is allocated to the truck j and is worth 0 otherwise.

Major headings should be typeset in boldface with the first letter of important words capitalized.

2.1. Variables of problem

For any reservation i and any truck j, we consider:

- X_{ij}: boolean decision variable which is worth 1 if the reservation i must be allocated to the truck j and is worth 0 Otherwise;
- d_{ij}: distance between the reservation i and the truck j;
- δ_j : total distance traveled by the truck j after its last preventive systematic maintenance (draining, control,…);
- Ldmj: maximal distance not to be exceeded between two systematic preventive maintenances of the truck j;
- dtg_j: total distance of the operation of the truck j until return in the garage for preventive maintenance (if necessary);
- ε: weight attributed to the objective function allowing to disadvantage the cost engendered by the unbalanced of load of trucks with regard to the cost of the fuel.
- m: number of available trucks;
- n: number of reservations.

2.2. Constraints

The various constraints of our problem are:

- A reservation i must never be denied;
- A truck j must be assigned to one and only one reservation;

$$\sum_{i=1}^{n} X_{ij} \leq 1 \forall j \in \{1,...,m\} \quad (1)$$

- A reservation i must be assigned to one and only one truck;

$$\sum_{j=1}^{m} X_{ij} = 1 \forall i \in \{1,...,n\} \quad (2)$$

- A truck can be affected only if it can insure the whole trip, i.e. if there will be any maintenance to do during the trip that has been assigned.

$$\delta_j + dtg_j \leq Ldm_j \quad (3)$$

2.3. Costs

The costs to be minimized, occurring in the objective function, are the cost of the fuel consumed during the travel and the cost engendered by the imbalance of load during these trips.

2.3.1. Cost of fuel

This cost is expressed in terms of distances traveled. Its formula as following (§ Eq. (4)):

$$\sum_{i=1}^{n}\sum_{j=1}^{m} d_{ij} X_{ij} \quad (4)$$

2.3.2. Cost of load imbalance of trucks

Without decision-making tool, the person in charge of assignment of truck affects at present the reservations in a random way. This method doesn't unfortunately consider the distance already traveled by trucks. So, a truck can be assigned several times and aging faster than planned. Consequently, the forecasts of the preventive maintenance can be erroneous because certain trucks can use their potential faster or slower than the rest of the fleet. Therefore, the assignment should be based on the history of each truck mileage to assign the trucks drove less than the others. So, the cost engendered by the imbalance of the load for the fleet is as following (§ Eq. (5)):

$$\sum_{i=1}^{n}\sum_{j=1}^{m} \delta_j X_{ij} \quad (5)$$

Otherwise, both cost of the objective function does not have the same importance, we assign a weight ε to the second component of the objective function to disadvantage the cost caused by the imbalance of load with regard to to the cost of fuel. ε is selected such

that the main objective is the minimization of the distance between the truck i and the booking j [Afsar (2007)].
Finally, the objective function can be expressed as following:

$$Min((\sum_{i=1}^{n}\sum_{j=1}^{m} d_{ij}X_{ij}) + (\varepsilon\sum_{i=1}^{n}\sum_{j=1}^{m}\delta_j X_{ij})) \tag{6}$$

Or:

$$Min\sum_{i=1}^{n}\sum_{j=1}^{m}(d_{ij}+\varepsilon\delta_j)X_{ij} \tag{7}$$

2.4. Formulation of problem

We are thus in the presence of a linear program with Boolean variables [Faure (2005)] [Boussier (2008)] which can be express as following (§ Eq. (8)):

$$(S)\begin{cases} Min\sum_{i=1}^{n}\sum_{j=1}^{m}(d_{ij}+\varepsilon\delta_j)X_{ij} \\ \sum_{i=1}^{n} X_{ij} \le 1 \\ \sum_{j=1}^{m} X_{ij} = 1 \\ \delta_j + dtg_j \le Ldm_j \\ X_{ij} \in \{0,1\} \\ \forall i \in \{1,...,n\} \forall j \in \{1,...,m\} \end{cases} \tag{8}$$

3. Resolution of the Problem of Assignment of trucks with an Exact Method (ATEM)

We propose for the resolution of the problem an exact method based on the linear programming with boolean variables using the solver LP-SOLVE Version5 [a].
The obtained optimal solution is $(X_{ij})_{\substack{1\le i\le n \\ 1\le j\le m}}$ where every X_{ij} is worth 1 if the reservation i must be assigned to the truck j and 0 otherwise.

Table 1. analysis of the results of the solution proposed by our model ATEM.

Number of Reservations	**Optimal solution**	
	solution (Km)	**CPU(s)**

30	5653,5	2
50	8397,4	3s70
80	14654,41	32s4
100	17374,23	51s45
130	22470,24	69s23
150	20391,4	80s09
200	34546,02	172s34
224	29562,8	247s40
260	44657,12	450s12
300	53765,9	637s

We proceeded with various simulations with real data given by the trucking company (§Table. 1), these simulations show, firstly, that the solution obtained by this method in a short computation time is optimal, and secondly, that this solution minimizes the total distance traveled by the truck while taking account of the problem of balancing of these vehicles.

4. Ant Colony Optimization

Ant Colony Optimization (ACO) [Stutzle (1997)] [Dorigo et al. (2004)] [Liao et al. (2011)] is a metaheuristic for solving hard combinatorial optimization problems. The inspiring source of ACO is the pheromone trail laying and following behavior of real ants, which use pheromones as a communication medium. In analogy to the biological example, ACO is based on indirect communication within a colony of simple agents, called (artificial) ants, mediated by (artificial) pheromone trails. The pheromone trails in ACO serve as distributed, numerical information, which the ants use to probabilistically construct solutions to the problem being solved and which the ants adapt during the algorithm's execution to reflect their search experience.

The (artificial) ants in ACO implement a randomized construction heuristic which makes probabilistic decisions as a function of artificial pheromone trails and possibly available heuristic information based on the input data of the problem to be solved. As such, ACO can be interpreted as an extension of traditional construction heuristics, which are readily available for many combinatorial optimization problems.

Ants are wandering into a random direction in order to find food. A trail of pheromones is left behind so the ant can track its way back when the insect has found food. On the way back the insect is distributing another layer of pheromone on the path. Other ants sense the pheromones and follow the trail by a certain probability. The density of the pheromones on the path increases and more ants are likely to follow the path to the food. With a higher concentration of pheromones on the trail, ants are more likely to stay on the trail, as can be seen from Fig.2. The pheromones vaporize in time and when all the food is collected the pheromone trail is no longer renewed and the ants head to new locations in random directions.

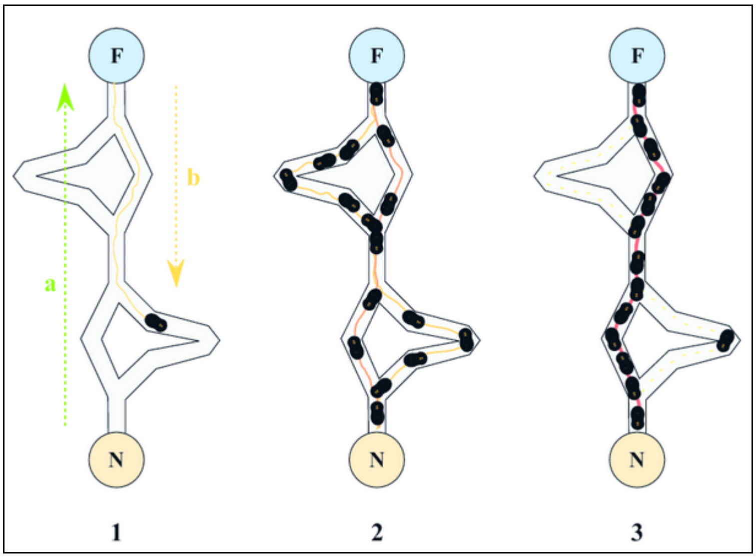

Fig 2. Ants finding the shortest path from the food (F) to the nest (N) by following pheromones

4.1. Constants of the problem

We defined below the various constants of the problem:

- f : number of ants. We shall identify the ants and their solutions, stored in a table sorted out by decreasing distance. The best solution is thus the last one (index f)
- I_{max} : number of iterations
- η_{ij} : a visibility parameter to city j from city i with $\eta_{ij} = 1/d_{ij}$
- ρ *:* evaporation rate of pheromone with $0 \leq \rho \leq 1$.

  ρ is a coefficient which determines the speed of evaporation of pheromone on the arcs, if ρ=1 there is no evaporation and if ρ = 0, the ants take into account only the deposits of pheromone of the last cycle.
- Q: Adjustment parameter to calculate the quantity of pheromone deposited between time t and t +1.

4.2. Variables of the problem

The variables of the problem are:

- τ_{ij}^{k} :quantity of pheromone on the arc (i,j).
- $\Delta\tau_{ij}^{k}$:addition of pheromone by the ant k on the arc (i,j).
- L_k :list "taboo" of the ant k (all solutions already found by the ant k)
- P_{ij}^{k} :probability that ant k moves the city i to city j.

4.3. Ant System

Ant System (AS) is the first ACO algorithm reported in the literature [Dorigo et al.(1991)] [Dorigo et al. (1996)]. In AS, an ant k being in node i choose the next node j with a probability given by the random proportional rule defined as:

$$p_{ij}^{k}(t) = \begin{cases} \frac{\tau_{ij}(t)^{\alpha} \cdot \eta_{ij}^{\beta}}{\sum_{l \in J_i^k} \tau_{il}(t)^{\alpha} \cdot \eta_{il}^{\beta}} & \text{si } j \in J_i^k \\ 0 & \text{si } j \notin J_i^k \end{cases} \tag{9}$$

Where:

- J_i^k is the list of the possible displacements for the ant k when she is on a city i
- $\tau_{ij}(t)$ the intensity of the tracks at a given iteration t.
- α et β are two positive factors that control the relative importance of intensity and visibility, these factors have a decisive influence on the intensification and diversification of the search as we will see later.

The second step of Ant system algorithm is the update of pheromone traces. During the execution of this step, for any arc process (i,j), the initial quantity of pheromone on arcs is a uniform distribution of a small positive quantity $\tau_{ij}(0)=\tau_0$. Once the tour is done, an ant k deposits a quantity of pheromone $\Delta\tau_{ii}^{k}$ on each arc of its tour (formula. (10)):

$$\Delta\tau_{ij}^{k} = \begin{cases} \frac{Q}{L^k(t)} si(i,j) \in T^k(t) \\ 0 si(i,j) \notin T^k(t) \end{cases} \tag{10}$$

Where T^k (t) is the tour constructed by ant k at the iteration t and $L^k(t)$ is the length of this tour. Q is a parameter of regulation.

At the end of each iteration, all m ants deposit pheromone. The pheromone trail values are the sum of pheromones which did not evaporate and of those who have just been deposited (§formula. (11)):

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{i=1}^{f} \Delta\tau_{ij}^{k}(t) \tag{11}$$

4.4. Max-Min Ant System

An effective variant of ACO is Max-Min Ant System MMAS ([Stutzle (1997)], [Stutzle (2000)]) where only the best draw tracks and ants deposit pheromone which is limited by an upper bound (preventing a track from being too much strengthened) and a lower bound (leaving the possibility to be explored in any solution), the initial value of pheromone τ_0 is fixed to τ_{max}. This algorithm achieves better results than the original, and avoids a premature convergence. The algorithm ATAC which we propose to solve our problem of assignment is based on the principle of Max-Min ant system and is defined as (§Fig. 3):

Initialization of the parameters (f, α, β, ρ,Q, I_{max})
Initialize pheromone trails to τ_{max} for every arc (i, j) $\tau_{ij}(0)= \tau_{max}$
Affect f ants in m cities of reservation
Repeat
For each city of reservation i ranging from 1 to m **do**
 For each ant k ranging from 1 to f **do**
  For each truck j ranging from 1 to n **do**
   Calculate the probability P_{ij}^{k} according to the formula (9)
  End for
  Move the kth ant towards the city of reservation j according to the best found probability P_{ij}^{k}
  Calculate the quantity of pheromone $\Delta\tau_{ij}^{k}$ deposited by the ant k according to (10)
 End for
 Update the tracks of pheromone τ_{ij} according to the formula (11)
 If a track of pheromone is below τ_{min} then set it in τ_{min}
 If a track of pheromone exceeds τ_{max} then set it in τ_{max}
End for
Sort out the solutions by decreasing distance
Retain the good solution
Until reach I_{max} iterations

Fig 3. Pseudo-code of ATAC algorithm.

5. Implementation and Simulations

The program used for our experiments was developed with Java language. We chose this language because it can be implemented on many different types of computers (cell phone, mac, PC, linux, etc.), that allows to install our application on the cell phones and the tablets pc of the logistic managers to insure the speed of treatment of reservations, on the other hand the other languages (C, C++, ...) can only be run on a computer of the same type as the one that compiled the program.
These simulations were conducted on a Pentium IV 3.2 Ghz / 4GB RAM.

5.1. Regulation of parameters

5.1.1. Parameters α and β

The choice of the parameters α and ß has a very strong influence on the process of construction of the solutions, he allows counterbalancing the algorithm in a phase of intensification or diversification. The trap maladjustment between intensification and diversification would rapidly converge in a local minimum (too strong intensification corresponds to α very big) or never be solved (too strong diversification corresponds to β very big).

The tested values are $\alpha \in \{1, 2, 3\}$ and $\beta \in \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$. After several simulations, the minimal value of the total distance traveled is obtained when α et β are worth respectively 1 and 3 (§Table.2).

Table 2. Regulation of the parameters α and β.

	N° test	Number of Reservations	**β**								
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
			sol	sol	sol	sol	sol	sol	sol	sol	sol
α = 1	1	30	5748,1	5663,4	5657,9	5654,4	5662,7	5884,2	5756,0	6178,9	6436,8
	2	50	8475,7	8412,4	8408,4	8399,6	8399,9	8402,7	8456,7	8614,2	8624,1
	3	150	21090,7	20948,1	20900,5	20909,7	20945,2	20940,8	21011,4	21260,5	21390,4
	4	224	31812,6	31806,4	31602,2	31602,9	31650,1	31834,0	31850,7	33845,9	34369,4
	5	300	47589,4	45325,4	43691,7	42890,1	42458,6	44265,7	46258,4	49521,0	49325,7
	6	400	63258,4	61256,2	60758,4	61760,1	62856,4	63452,1	65954,1	66485,9	68568,5
	Average		29662,5	28902,0	**28503,2**	28536,1	28662,2	29129,9	29947,9	31001,1	31452,5
	Min (Average)		**28503,2**								
α = 2	1	30	6663,9	5896,4	5836,0	5790,7	5806,8	6063,7	6100,5	6401,7	6638,4
	2	50	9123,0	8753,5	8752,1	8736,7	8546,5	8493,6	8602,9	8932,3	8965,3
	3	150	24364,9	22150,2	22050,6	21290,5	23658,2	23968,6	25632,7	28639,4	28962,4
	4	224	35652,7	33853,1	32963,1	32602,2	33653,1	33968,1	35692,7	36965,3	36999,4
	5	300	45963,4	44569,2	42658,6	42750,9	43025,1	43056,0	43682,1	44563,2	45963,2
	6	400	65321,5	64369,0	63558,4	64326,1	66452,3	66985,4	67653,1	69987,2	70236,1
	Average		31181,6	29931,9	29303,1	**29249,5**	30190,3	30422,6	31227,3	32581,5	32960,8
	Min (Average)		**29249,5**								
α = 3	1	30	6123,8	5990,0	5963,8	5870,1	5968,6	6158,4	6298,2	6452,3	6756,1
	2	50	9263,4	8963,1	8793,4	8745,6	8693,6	8796,1	8963,7	9236,4	9354,0
	3	150	24563,8	23695,1	21970,3	23653,7	24563,8	26353,8	26357,2	28635,1	29685,0
	4	224	33869,1	33152,3	32865,0	34045,1	34582,2	35693,1	35989,0	37458,5	38654,1
	5	300	44256,7	43652,7	42954,4	43695,1	44756,3	45856,3	45963,1	46123,0	47023,1
	6	400	66521,2	65458,2	64423,8	63365,7	65345,9	66963,0	67401,1	69908,5	69012,3
	Average		30766,3	30151,9	**29495,1**	29895,9	30651,7	31636,8	31828,7	32969,0	33414,1
	Min (Average)		**29495,1**								

5.1.2. Parameter ρ

The rate of evaporation of pheromones is one of the main parameters of the algorithm. If the evaporation rate is too small (or too big), the algorithm is penalized. When the rate is too small, pheromones accumulate and we take the risk of being trapped in a local

minimum. With a rate too big, the ants don't have the time to take advantage of the information gathered by their colleagues that already pheromones are evaporated. This means exactly to eliminate the collective knowledge, which is precisely the strength of the method of ant colonies. The algorithm does not converge (or very slowly).
We fixed α in 1 and β in 3, and we looked the best value of ρ. the ρ's values tested are $\{0.1, 0.2, 0.3, 0.5, 0.9\}$. After several simulations, we found that the best value of ρ is 0.1 (§ Table3).

Table 3: Regulation of the parameter

Number of Reservations	ρ				
	0,1	0,2	0,3	0,5	0,9
30	5657,9	5796,5	57843,7	56963	6596,4
50	8408,4	8379,4	8456,9	8364,3	9409,1
150	20900,5	20640,1	23586,4	22356,1	26014,7
224	31602,2	30154,3	31856,1	320125,4	35330,4
300	43691,7	44785,4	45869,3	44250,1	49869,3
400	60758,4	61458,1	63125,6	64590,7	74325,9
Average	**28503,2**	28535,6	38456,3	86108,3	33591
Min (Average)	**28503,2**				

5.1.3. Stopping criterion: maximum number of iteration I_{max}

After several simulations, we find that the number of iterations needed to find the best solution depends on the size of the problem. Furthermore, these tests show that the algorithm stagnates at a certain number of iterations. Thus, our algorithm runs respectively 184, 204, 300, 484, 512, 1102 iterations on average to solve the assignments of size 30, 50, 150, 224, 300 and 400 reservations (§ Figure.3).

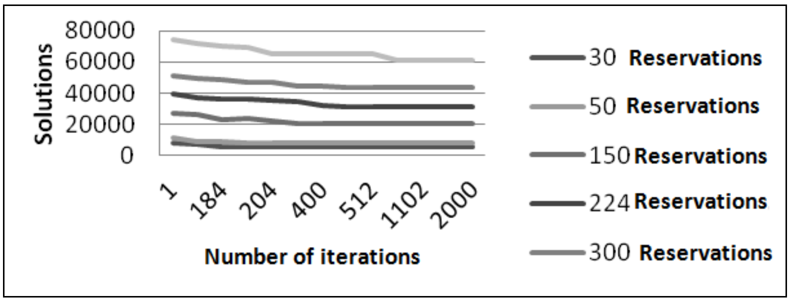

Fig 3. Evolution of the quality of solution with regard to the number of iterations.

Besides, We can notice that more the number of reservations increases more the number necessary for the algorithm to converge on a good solution grows: consequently the calculation time increases . So, according to the figure 3, we can fix the maximal number of iterations according to the table below.

Table4 : Maximal number of iterations according to number of reservations

Number of reservations	<= 50	<=1 50	<= 300	<= 400
Number of iterations	210	300	520	1110

5.1.4. Parameters f,Q, τ min et τ max

Several studies [Costanzo(2006)] proved that the optimum exists when the number of ants is equal to the number of cities, and it is more interesting to distribute ants on all the cities rather than to make them start of a city. Initially in our algorithm, we place an ant by city of reservation.

The parameter Q has only an unimportant influence because in the politics of choice of ants, we use a ratio of the quantities of pheromones and not directly these quantities. We put generally Q = 100 [Costanzo (2006)].

The parameters τ_{min} and τ_{max} are experimentally chosen, after several simulations we fix τ_{min} to 0.01 and τ_{max} in 4.

We summarize the best values found experimentally in table5:

Table 5: Parameters chosen for our algorithm ATAC

Parameter	**Value**
α	1
β	3
ρ	0,1
I_{max}	§Table. 4
f	n
Q	100
τ_{min}	0,01
τ_{max}	4

6. Comparison of Two Proposed Methods: Method ATEM AND ATAC

We noticed that our method ATEM gives optimal solutions at reasonable time for a number of reservations less than 300 reservations, over 300 reservations we use our algorithm ATAC which always proposes good solutions close to the optimum in an interesting calculation time. To ensure the quality of the results of our model ATAC in case we don't have optimal solutions, lower bounds (LB) were found by the software AIMMS (Asset and Inventory Management Systems software) "AIMMS" which is an optimization modeling, the table below shows that the solutions proposed by our method ATAC are close to the lower bound obtained, with a standard deviation varying between 5.4% and 6.3% (§Table.6). The standard deviation is calculated as follows:

Standard deviation = ((Solution ATAC - solution ATEM "or lower bound") / solution ATEM "or lower bound ") *100

To test the efficiency of both proposed methods, we proceeded to diverse simulations (§Table.6) with real data given by the road transport.

Table 6: comparison of results between method ATEM and ATAC in terms of total distance and processing time

N° test	Number of Reservations	ATEM		LB		ATAC		Standard deviation (%)
		solution (Km)	CPU(s)	solution (Km)	CPU(s)	solution (Km)	CPU(s)	
1	30	5653,5	2	-	-	5657,9	12	0,08
2	50	8397,4	3s70	-	-	8408,4	17s40	0,13
3	150	20391,4	80s09	-	-	20900,5	63s13	2,5
4	224	29562,8	247s40	-	-	31602,2	150s9	6,9
5	300	41934,8	637s	-	-	43691,7	393s01	4,2
6	400	-	-	57621,7	3004s2	60758,4	514s10	5,4
7	450	-	-	61121,2	4214s1	65019,7	607s3	6,3
8	500	-	-	67455,7	4042s6	71641,6	711s58	6,2

The results allow concluding that both our proposed methods are effective for solving the problem of assigning of trucks

- ❖ The ATEM method proposes optimal solutions for problems of size less than 300 reservations.
- ❖ The ATAC method proposes very good solutions close to optimal solutions in acceptable calculation time and with an average standard deviation of 2.76% with regard to the optimum and 6% compared to the lower bounds (number of reservations upper to 300).

7. Conclusion

After a detailed study of the existing, we modeled the problem of our road transport and to propose a decision-making tool which allows the persons in charge a better assignment of trucks while minimizing the total distance traveled and by respecting the various constraints: the satisfaction of all the demands of the customers, the balancing of the loads of trucks and their preventive maintenances.

We also managed to adapt dynamically and automatically the parameters of our algorithm ATAC based on the principle of ant colony. Our goal was to find good results at acceptable time by the user. The experimental results obtained with the values chosen

are very encouraging and well meet the requirements of our trucking company
Our decision-making tool proposes optimal solutions for problems of size less than 300 reservations, and very good solutions close to optimal solutions in a reasonable calculation time when the number of reservations exceeds 300.

References

Afsar, M.(2007) : Affectation des vols aux appareils d'une compagnie aérienne avec prise en compte des contraintes de maintenance. «doctoral thesis », INPG, France.

Boussier, S.(2008) : Etude de résolution Search pour la programmation linéaire en variables binaires «doctoral thesis». Université Mentpellier II.

Costanzo, A., Luong, T., Marill, V.(2006) : Optimisation par colonies de fourmis.Rapport, université Nice Sophia-Antipolis.

Dorigo, M.; Maniezzo, V.; Colorni, A. (1991): The ant system: an autocatalytic optimizing process.Italy: Dipartimento di Elettronica, Politecnico di Milano. pp. 91–016.

Dorigo, M.; Maniezzo, V.; Colorni, A. (1996):. Ant System: optimization by a colony of cooperating agents. IEEE Trans Syst Man and Cybern-Part B;26(1):29–41.

Dorigo, M., Stützle, T.(2004): Ant colony optimization. MIT Press.

Faure, R.(2005) : Recherche opérationnelle et programmation linaire.

Grellier, É.(2008): Optimisation des tournées de véhicules dans le cadre de la logistique inverse : modélisation et résolution par des méthodes hybrides. «doctoral thesis », Université de Nantes, France.

Iori, M., Salazar-Gonzalez, J., Vigo, D.(2007): An exact approach for the vehicle routing problem with two-dimensional loading constraints. Transportation Science, 41, 253-264.

Khebbache-Hadji, S., Prins, C., Alaoui,A. (2010); Problèmes de tournées de véhicules avec chargement et fenêtres horaires. 8e Conférence Internationale de MOdélisation et SIMulation (MOSIM'10), Hammamet – Tunisie.

Khichane, M.(2009) : Un modèle réactif pour l'optimisation par colonies de fourmis : application à la satisfaction de contraintes. Actes JFPC.

Lacomme, P., Prins, C., Tanguy, A. (2010) : Optimisation par colonies de fourmis pour les tournées sur arcs. 8e Conférence Internationale de MOdélisation et SIMulation (MOSIM'10), Toulouse (France).

Liao, T., de Oca ,M., Aydin ,D., Stützle ,T., Dorigo ,M.(2011): An incremental Ant Colony Algorithm with Local Search for Continuous Optimization. In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2011), pages 125-132. ACM Press, New York.

Rego, C. (1994) : Le problème de tournées de vehicules : Étude et résolution approchée. Technical Report 2197, INRIA - Institut National de Recherche en Informatique et en Automatique.

Solnon, C. (2010) : Résolution de problèmes combinatoires et optimisation par colonies de fourmis.

Stutzle, T., Hoos, H.(1997): Improvements on the Ant System : Introducing MAX-MIN Ant System. In Proceedings International Conference on Artificial Neural Networks and Genetic Algorithms, Vienna. Springer-Verlag.

Stutzle, T., Hoos, H.(2000): MAX-MIN Ant System. Future Generation Computer System, 16:889.

Notes de cours. URL http://www710.univ-lyon1.fr/

[a]http://lpsolve.sourceforge.net.