Global Scientific JOURNALS

GSJ: Volume 9, Issue 7, July 2021, Online: ISSN 2320-9186
www.globalscientificjournal.com

ON MAXIMIZATION OF PROFITS IN SOME ESTABLISHMENTS USING ANT COLONY OPTIMIZATION

[1]Oke, M.O., [2]Elegbede, O.O., [3]Raji, R.A., [4]Adenipekun, A.E. and [5]Adetolaju, O.S.
[1,2]Department of Mathematics, Ekiti State University, Ado-Ekiti, Nigeria
[3]Department of Mathematics & Statistics, Osun State Polytechnic, Iree, Nigeria
[4]Department of Statistics, Federal Polytechnic Ede, Osun State, Nigeria
[5]Department of Computer Science, Ekiti State University, Ado-Ekiti, Nigeria

ABSTRACT:

Many problems of combinatorial optimization arise in engineering, science, management and social sciences in which a set of optimal solution is required for a function represented as a tuple $\rho = (S, F)$. Ant colony optimization is an heuristics optimization technique for solving combinatorial optimization problems. In this paper, a metaheuristic algorithm is given to solve unconstrained optimization problems involving maximization of profits in some establishments. The algorithm was tested on four classes of problems from different establishments with paths ranging from 9 to 41 and ants ranging from 4 to 20. The superiority of Ant colony optimization over other methods was shown when we compared our results with the solutions obtained using Fibonacci search method as it performs better in all the problems considered.

KEYWORDS: Ant Colony Optimization, Chemical Pheromone Trails, Combinatorial Optimization, Fibonacci search method, Swarm Intelligent, Turple.

1.0 INTRODUCTION

Optimization is the act of obtaining the best result under any given circumstances. Engineers and scientists normally take many technological and managerial decisions at several stages of design, construction, maintenance and managerial systems. The ultimate goals of such decisions are to either minimize the efforts required or maximize the desired benefits, [11, 12]. Since the efforts required or the benefits desired in any practical situation can be expressed as functions of certain decision variables, optimization can be defined as the process of finding the conditions that give the maximum or minimum value of a function, [12].
Modern optimization techniques, sometimes called nontraditional optimization methods, have emerged as powerful and popular methods for solving complex optimization problems in recent years, [9, 10]. Ant colony optimization (ACO) is a technique of optimization that was introduced in the early 1990’s by Dorigo for solving combinatorial optimization problems. The inspiring source of ACO is the foraging behaviour of real ant colonies. This behaviour is exploited in

artificial ant colonies for the search of approximate solutions to discrete and continuous optimization problems, [3, 4, 6]. The agents, called ants, are very efficient at sampling the problem space and quickly finding good solutions to it. Ant algorithms are multi-agent systems in which the behaviour of each single agent, called artificial ant, is inspired by the behaviour of real ants, [1, 3, 5]. At the core of this behaviour is the indirect communication between the ants by means of chemical pheromone trails, which enable them to find short paths between their nest and food sources, [3, 4]. This characteristic of real ant colonies is exploited in ACO algorithms in order to solve optimization problems, [4, 7].
A combinatorial optimization problem can be represented as a tuple $\rho = (S, F)$, where S is the solution space with $s \in S$ been a specific candidate solution and $F: S \to \mathbb{R}_+$ a fitness function assigning strictly positive values to candidate solutions, [3, 13]. In this case, higher values correspond to better solutions $S^* \subseteq S$ that maximize the fitness function. The solution s^* is then called an optimal solution and S^* is called the set of optimal solutions.

A lot of researches had been carried out on this modern or nontraditional method of optimization. Stützle and Hoos [14] developed an algorithm which gave some improvements on ant system while Stützle and Darigo [13] gave some convergence proofs for a class of ant colony optimization algorithms. Dorigo and Gambardella [4], Gambardella and Dorigo [7] and Lawler *et al*. [8] applied the modern methods of optimization to solve the traveling salesman problems. Dorigo *et al*. [5], Oke [9] and Oke [10] applied some modern optimization techniques to solve complex engineering problems.
We are therefore motivated by the fact that none of these researchers have worked on the application of ACO algorithms in maximizing profits. Therefore, in this research, we considered the maximization of profits in some establishments using ant colony optimization techniques.

2.0 MATERIALS AND METHODS

In applying the ACO algorithm to unconstrained optimization problems for the maximization of profits in some establishments, we considered a profit function of the form:

Profit Function = Revenue Function - Cost Function, [2]

That is

$$P(X) = R(X) - C(X) \tag{1}$$

where $P(X)$ = Profit function

$R(X)$ = Revenue function

$C(X)$ = Cost function, [2]

In the maximization of profits in the establishment considered, the following steps were followed in applying the ACO algorithm for the solution of the problems:

Step 1: Assume a suitable number of ants in the colony (N) and a set of permissible discrete values for each of the n design variables. Denote the permissible discrete values of the design variable x_i as $x_1, x_2, \ldots, x_{ip}$ $(i = 1,2,\ldots,n)$. Assume equal amounts of pheromone $\tau_{ij}^{(l)}$ initially along all the arcs or rays (discrete values of design variables) of the multilayered graph. The superscript to τ_{ij} denote the iteration number and for simplicity, let $\tau_{ij}^{(l)} = 1$ be assumed for all arcs ij and set the iteration number $l = 1$.

Step 2: (a) Compute the probability (p_{ij}) of selecting the arc or ray (or the discrete value) x_{ij} as

$$p_{ij} = \frac{\tau_{ij}^{(l)}}{\sum_{m=1}^{p} \tau_{im}^{(l)}}; i = 1,2,3, \ldots \ldots, n; j = 1,2,3, \ldots \ldots. p \quad (2)$$

(b) The specific path (or discrete values) chosen by the kth ant can be determined using random numbers generated in the range (0, 1). For this, we find the cumulative probability ranges associated with different paths based on the probabilities given by equation (2). The specific path chosen by ant k will be determined using the roulette-wheel selection process in step 3(a).

Step 3: (a) Generate N random numbers $r_1, r_2, \ldots, r_N$ in the range (0, 1), one for each ant. Determine the discrete value or path assumed by ant k for variable I as the one for which the cumulative probability range [found in step 2(b)] includes the value r_i.

(b) Repeat step 3(a) for all design variables $i = 1,2,\ldots,n$.

(c) Evaluate the objective function values corresponding to the complete paths (design vectors $X^{(k)}$ or values of x_{ij} chosen for all design variables $i = 1,2,\ldots,n$ by ant k, $k = 1,2,\ldots,N$):

$$f_k = f\,(X^{(k)});\; k = 1,2,\ldots,N \quad (3)$$

Determine the best and worst paths among the N paths chosen by different ants:

$$f_{best} = \min_{k=1,2,\ldots,N} \{f_k\} \quad (4)$$

$$f_{worst} = \max_{k=1,2,\ldots,N}\{f_k\} \tag{5}$$

Step 4: Test for the convergence of the process. The process is assumed to have converged if all N ants take the same best path. If convergence is not achieved, assume that all the ants return home and start again in search of food. Set the new iteration number as $l = l + 1$, and update the pheromones on different arcs (or discrete values of design variables) as

$$\tau_{ij}^{(l)} = \tau_{ij}^{(old)} + \sum_k \Delta\tau_{ij}^{(k)} \tag{6}$$

where $\tau_{ij}^{(old)}$ denotes the pheromone amount of the previous iteration left after evaporation, which is taken as

$$\tau_{ij}^{(old)} = (1-\rho)\tau_{ij}^{(l-1)} \tag{7}$$

and $\Delta\tau_{ij}^{(k)}$ is the pheromone deposited by the best ant k on its path and the summation extends over all the best ants k (if multiple ants take the same best path). Note that the best path involves only one arc i, j (out of p possible arcs) for the design variable i. The evaporation rate or pheromone decay factor ρ is assumed to be in the range 0.5 to 0.8 and the pheromone deposited $\Delta\tau_{ij}^{(k)}$ is computed using the equation

$$\Delta\tau_{ij}^{(k)} = \begin{cases} \dfrac{\zeta f_{best}}{f_{worst}}; & if\ (i,j) \in global\ best\ tour \\ 0; & otherwise \end{cases} \tag{8}$$

with the new values of $\tau_{ij}^{(l)}$, go to step 2. Steps 2, 3, and 4 are repeated until the process converges, that is, until all the ants choose the same best path, [3 – 6].

3.0 COMPUTATIONAL EXAMPLES

Problem 1: Maximize $f(x)$ in the range $220 \leq x \leq 300$

where $f(x)$ is the profit function of a firm given by:

$$f(x) = -200x^2 + 92000x - 8400000$$

Problem 2: Maximize $f(x)$ in the range $4700 \leq x \leq 6500$

where $f(x)$ is the profit function of a company given by:

$f(x) =- 0.000004x^3 + 0.025x^2 + \ 100x \ - 75000$

Problem 3: Maximize $f(x)$ in the range $150 \leq x \leq 250$

where $f(x)$ is the profit function of a firm given by:

$f(x) = -8x^2 + 3200x - 80000$

Problem 4: Maximize $f(x)$ in the range $0 \leq x \leq 80$

where $f(x)$ is the profit function of an industry given by:

$f(x) = -2x^2 + 200x - 2000$

4.0 RESULTS AND DISCUSSIONS

The solution to the problems using Ant Colony Optimization technique and the comparison between Ant Colony Optimization and Fibonacci Search methods were presented in this section.

Table 1: Ant Colony Optimization Method for Problem 1

$Ant \ = \ 4, n = 1, x = x_1,$ is assumed within the range of x_1 as $(p = 9) \ x_{ij} \ ; j = 1,2, \dots ,9$

$f(x) = -200x^2 + 92000x - 8400000,$ in the range $220 \leq x \leq 300$

Iterations	Number of ants on best path	x_{best}	x_{worst}	f_{best}	f_{worst}
1	1	$x_{13} = 240$	$x_{18} = 290$	2160000	1460000
2	1	$x_{12} = 230$	$x_{17} = 280$	2180000	1680000
3	3	$x_{12} = 230$	$x_{17} = 280$	2180000	1680000
4	4	$x_{12} = 230$		2180000	

Table 2: Fibonacci Search Method for Problem 1

$a = x_1 = 300, \quad b = x_3 = 220, \qquad n = 20, \quad \varepsilon = 0$

$f(x) = -200x^2 + 92000x - 8400000,$ in the range $220 \leq x \leq 300$

Itera-	f_n	f_{n-1}	x_1	x_3	x_2	x_4	$f(x_2)$	$f(x_4)$

tions								
1	10946	6765	300	220	250.557281	269.442719	2095479.7	1868854.39
2	6765	4181	250.5572812	220	231.671843	238.885438	2179441.0	2164209.8
3	4181	2584	231.6718425	220	224.458247	227.213595	2173859.8	2178447.19
4	2584	1597	231.6718425	227.213595	228.916494	229.968944	2179765.2	2179999.81
5	1597	987	231.6718425	229.968944	230.619394	231.021393	2179923.3	2179791.35
6	987	610	230.6193935	229.968944	230.217950	230.370988	2179990.6	2179972.47
7	610	377	230.2173497	229.968944	230.063827	230.122467	2179999.2	2179997
8	377	233	230.0638267	229.968944	230.005186	230.027585	2180000	2179999.85
9	233	144	230.0051855	229.968944	229.982787	229.991342	2179999.9	2179999.98
10	144	89	230.0051855	229.991342	229.996630	229.999898	2179999.9	2180000
11	89	55	230.0051855	229.999898	230.001918	230.003166	2180000	2180000

Table 3: Ant Colony Optimization Method for Problem 2

$Ant = 20, n = 1, x = x_1,$ is assumed within the range of x_1 as $(p = 41)\ x_{ij}\ ; j = 1,2,\ldots,41$

$f(x) = -0.000004x^3 + 0.025x^2 + 100x - 75000,$ in the range $4700 \leq x \leq 6500$

Iterations	Number of ants on best path	x_{best}	x_{worst}	f_{best}	f_{worst}
1	1	$x_{1,22} = 5645$	$x_{1,1} = 4700$	566615.7805	531958
2	1	$x_{1,22} = 5645$	$x_{1,41} = 6500$	566615.7805	532750
3	6	$x_{1,22} = 5645$	$x_{1,40} = 6455$	566615.7805	536333.04
4	17	$x_{1,22} = 5645$	$x_{1,5} = 4880$	566615.7805	543502.912
5	20	$x_{1,22} = 5645$		566615.7805	

Table 4: Fibonacci Search Method for Problem 2

$a = x_1 = 6500, \quad b = x_3 = 4700, \qquad n = 20, \quad \varepsilon = 0$

$f(x) = -0.000004x^3 + 0.025x^2 + 100x - 75000,$ in the range $4700 \leq x \leq 6500$

Itera-tions	f_n	f_{n-1}	x_1	x_3	x_2	x_4	$f(x_2)$	$f(x_4)$
1	10946	6765	6500	4700	5387.53883	5812.46117	563887.607	565374.598
2	6765	4181	6500	5812.46117	6075.07763	6237.38354	558330.901	550701.725
3	4181	2584	6075.07763	5812.46117	5912.77174	5974.76706	563436.347	561777.593
4	2584	1597	5912.77174	5812.46117	5850.77639	5874.45652	564741.866	564284.531
5	1597	987	5850.77693	5812.46117	5827.09629	5836.14128	565148.497	564999.142
6	987	610	5827.09629	5812.46117	5818.05213	5821.50618	565290.502	565237.128
7	610	377	5818.05128	5812.46117	5814.59641	5815.91605	565342.806	565322.954

8	377	233	5814.59641	5812.46117	5813.27676	5813.78083	565362.503	565354.998
9	233	144	5813.27676	5812.46117	5812.77270	5812.96522	565369.985	565367.130
10	144	89	5812.77270	5812.46117	5812.58016	5812.73227	565372.837	565370.584
11	89	55	5812.58016	5812.46117	5812.50663	5812.53471	565373.925	565373.510
12	55	34	5812.50663	5812.46117	5812.47853	5812.48927	565374.341	565374.182
13	34	21	5812.47853	5812.46117	5812.46781	5812.47189	565374.500	565374.439
14	21	13	5812.46781	5812.46117	5812.46370	5812.47192	565374.561	565374.438
15	13	8	5812.46370	5812.46117	5812.46215	5812.46273	565374.584	565374.575
16	8	5	5812.46215	5812.46117	5812.46154	5812.46178	565374.593	565374.589
17	5	3	5812.46154	5812.46117	5812.46132	5812.46139	565374.596	565374.595
18	3	2	5812.46132	5812.46117	5812.46122	5812.46127	565374.597	565374.597
19	2	1	5812.46122	5812.46117	5812.46120	5812.46120	565374.598	565374.598

Table 5: Ant Colony Optimization Method for Problem 3

$Ant = 8, n = 1, x = x_1$, is assumed within the range of x_1 as $(p = 11)\ x_{ij}$; $j = 1,2,\dots,11$

$f(x) = -8x^2 + 3200x - 80000$, in the range $150 \leq x \leq 250$

Iterations	Number of ants on best path	x_{best}	x_{worst}	f_{best}	f_{worst}
1	1	$x_{1,6} = 200$	$x_{1,1}, x_{1,11} = 150,250$	240000	220000
2	3	$x_{1,6} = 200$	$x_{1,1} = 150$	240000	220000
3	6	$x_{1,6} = 200$	$x_{1,9} = 230$	240000	232800
4	7	$x_{1,6} = 200$	$x_{1,2} = 160$	240000	227200
5	8	$x_{1,6} = 200$		240000	

Table 6: Fibonacci Search Method for Problem 3

$a = x_1 = 250, \quad b = x_3 = 140, \quad n = 20, \quad \varepsilon = 0$

$f(x) = -8x^2 + 3200x - 80000$, in the range $150 \leq x \leq 250$

Itera-tions	c_n	f_{n-1}	x_1	x_3	x_2	x_4	$f(x_2)$	$f(x_4)$
1	10946	6765	250	140	182.016262	207.983738	237412.681	239490.079
2	6765	4181	250	207.983738	224.032522	233.951217	235379.503	230778.519
3	4181	2584	224.032522	207.983738	214.113829	217.902432	238406.399	237436.024
4	2584	1597	214.113829	207.983738	210.325224	211.772343	239147.118	238891.296
5	1597	987	210.325224	207.983738	208.878107	209.430856	239369.434	239288.472

6	987	610	208.878107	207.983738	208.325356	208.536489	239445.508	239417.027
7	610	377	208.325356	207.983738	208.114225	208.194870	239473.275	239462.753
8	377	233	208.114225	207.983738	208.033580	208.064384	239483.693	239479.726
9	233	144	208.033580	207.983738	208.002776	208.014542	239487.645	239486.137
10	144	89	208.002776	207.983738	207.991010	207.995505	239489.150	239488.575
11	89	55	207.991010	207.983738	207.986516	207.988232	239489.725	239489.505
12	55	34	207.986516	207.983738	207.984799	207.985456	239489.944	239489.86
13	34	21	207.984799	207.983738	207.984144	207.984394	239490.028	239489.996
14	21	13	207.984144	207.983738	207.983893	207.983990	239490.060	239490.047
15	13	8	207.983893	207.983738	207.983798	207.983834	239490.072	239490.067
16	8	5	207.983798	207.983738	207.983761	207.983776	239490.077	239490.075
17	5	3	207.983761	207.983738	207.983747	207.983752	239490.078	239490.078
18	3	2	207.983747	207.983738	207.983741	207.983744	239490.079	239490.079

Table 7: Ant Colony Optimization Method for Problem 4

$Ant = 8, n = 1, x = x_1$, is assumed within the range of x_1 as $(p = 17)\ x_{ij}$;

$j = 1,2, \ldots ,17$

$f(x) = -2x^2 + 200x - 2000$, in the range $0 \le x \le 80$

Iterations	Number of ants on best path	x_{best}	x_{worst}	f_{best}	f_{worst}
1	1	$x_{1,11} = 50$	$x_{14} = 15$	3000	550
2	14	$x_{1,11} = 50$	$x_{14} = 15$	3000	550
3	18	$x_{1,11} = 50$	$x_{1,12}, x_{1,10}$=55,45	3000	2950
4	20	$x_{1,11} = 50$		3000	

Table 8: Fibonacci Search Method for Problem 4

$a = x_1 = 80, \quad b = x_3 = 0, \qquad n = 20, \quad \varepsilon = 0$

$f(x) = -2x^2 + 200x - 2000$, in the range $0 \le x \le 80$

Itera-tions	f_n	f_{n-1}	x_1	x_3	x_2	x_4	$f(x_2)$	$f(x_4)$
1	10946	6765	80	0	30.5572812	49.4427188	2243.96137	2999.378875
2	6765	4181	80	49.44272	61.1145132	68.3281575	2752.93305	2328.157291
3	4181	2584	61.114561	49.44272	53.9009662	56.6563139	2969.56493	2969.564929
4	2584	1597	53.900966	49.44272	51.1456175	52.1980675	2997.37512	2990.337
5	1597	987	51.145618	49.44272	50.0931682	50.4951678	2999.98264	2999.509614

6	987	610	50.093169	49.44272	49.6911682	49.8447191	2999.80925	2999.951776
7	610	377	50.093169	49.84472	49.9396187	49.998269	2999.99271	2999.999994
8	377	233	50.093169	49.99827	50.0345171	50.0569204	2999.99762	2999.993524
9	233	144	50.034517	49.99827	50.0121149	50.0206713	2999.99971	2999.999145
10	144	89	50.012115	49.99827	50.0035574	50.0068265	2999.99998	2999.999907
11	89	55	50.003557	49.99827	50.0002893	50.0015371	2999.99999	2999.999999
12	55	34	50.003557	50.00154	50.0023085	50.0027860	2999.99998	2999.999998
13	34	21	50.002309	50.00154	50.0018320	50.0020135	2999.99998	2999.999999
14	21	13	50.002309	50.00201	50.0021259	50.0021961	2999.99998	2999.999992
15	13	8	50.002309	50.00220	50.0022393	50.0022652	2999.99998	2999.999994
16	8	5	50.002309	50.00227	50.0022814	50.0022922	2999.99999	2999.999991
17	5	3	50.002281	50.00227	50.0022717	50.0022275	2999.99998	2999.999988

Table 9: Comparison between ACO and Fibonacci Search Method for Problem 1

	ACO method	Fibonacci method
Function	$-200p^2+92000p-8400000$	$-200p^2+92000p-840000$
Range	$220 \le x \le 300$	$220 \le x \le 300$
x_{best}	230	230.001918, 230.0031656
f_{best}	2180000	2180000

Table 10: Comparison between ACO and Fibonacci Search Method for Problem 2

	ACO method	Fibonacci method
Function	$-75000+100x+0.025x^2-0.000004x$	$-75000+100x+0.025x^2-0.000004x^3$
Range	$4700 \le x \le 6500$	$4700 \le x \le 6500$
x_{best}	5645	5812.461198, 5812.461197
f_{best}	566615.7805	565374.5976

Table 11: Comparison between ACO and Fibonacci Search Method for Problem 3

	ACO method	Fibonacci method
Function	$-8x^2+3200x-80000$	$-8x^2+3200x-80000$
Range	$150 \le x \le 250$	$150 \le x \le 250$
x_{best}	200	207.9837414, 207.9837443
f_{best}	240000	239490.079

Table 12: Comparison between ACO and Fibonacci Search Method for Problem 4

	ACO method	Fibonacci method
Function	$200x - 2x^2 - 2000$	$200x - 2x^2 - 2000$
Range	$0 \leq x \leq 80$	$0 \leq x \leq 80$
x_{best}	**50**	**50.00227171, 50.00227496**
f_{best}	**3000**	**2999.999988**

5.0 CONCLUSION

The Ant Colony Optimization has been applied in the maximization of profits in some selected establishments. The results obtained showed the superiority of Ant Colony Optimization over the Fibonacci search method as it performs better in all the selected problems.

6.0 REFERENCES

.

[1] Bullnheimer, B., Hartl, R.F. and Strauss, C. (1999): A New Rank-based Version of the Ant System (A Computational Study), Central European Journal of Operations Research, 7(1), 25 - 38.

[2] Chiang, A. and Wainwright, K. (2013): Fundamental Methods of Mathematical Economics, Mc Graw-Hill Co. Inc., USA

[3] Darigo, M. and Blum, C. (2005): Ant Colony Optimization Theory (a Survey), Theoretical Computer Science, 344, 243 – 278.

[4] Dorigo, M. and Gambardella, L.M. (1997): Ant Colony System (A Cooperative Learning Approach to the Traveling Salesman Problem), IEEE Transaction on Evolutionary Computation, 1, 53 - 66.

[5] Dorigo, M., Maniezzo, V. and Colorni, A. (1991): The Ant System (An Autocatalytic Optimizing Process), Technical Report *TR91-016*, Politecnico di Milano.

[6] Dorigo, M. and Stützle, T. (2004): Ant Colony Optimization, Cambridge MA, MIT Press.

[7] Gambardella, L.M. and Dorigo, M. (1995): Ant-Q (A Reinforcement Learning Approach to the Travelling Salesman Problem), Proceedings of the 12th International Conference on Machine Learning, California, USA, 252 - 260.

[8] Lawler, E., Lenstra, J.K., Rinnooy, K.A. and Shmoys, D.B. (1985): The Travelling Salesman Problem, John Wiley & Sons, New York.

[9] Oke, M.O. (2012(a)): Solving Optimal Power Flow Problems using Partial Swarm Optimization Technique, Presented at the 23rd Annual Colloquium and Congress of the Nigerian Association of Mathematical Physics.

[10] Oke, M.O. (2012(b)): Differential Evolution Method of Solving Optimal Power Flow Problems, Presented at the 31st Annual Conference of the Nigerian Mathematical Society.

[11] Oke, M. O. (2014): A Second Order Method for Minimizing Unconstrained Optimization Problems, Journal of Applied Mathematics and Bioinformatics, International Scientific Press, UK, 4(3), 65 - 73.

[12] Rao, S.S. (1998): Optimization Theory and Application, Wiley Eastern Limited.

[13] Stützle, T. and Dorigo, M. (2002): A Short Convergence Proof for a Class of ACO Algorithms, IEEE Transactions on Evolutionary Computation, 6(4), 358 - 365.

[14] Stützle, T. and Hoos, H. (1997): Improvements on the Ant System. (Introducing MAX-MIN Ant System), Proceeding of International Conference on Artificial Neural Networks and Genetic Algorithms, Springer Verlag, 245 – 249.