Proceedings of the 6th WSEAS Int. Conf. on EVOLUTIONARY COMPUTING, Lisbon, Portugal, June 16-18, 2005 (pp188-192)

Ant Colony Optimization Algorithm (ACO); A new heuristic approach for engineering optimization

MOHAMMAD REZA JALALI, ABBAS AFSHAR
Department of Civil Engineering
Iran University of Science and Technology
Tehran, IRAN

MIGUEL A. MARINO
Department of Civil and Environmental Engineering
University of Calif., Davis
CA 95616, USA

Abstract: - Over the last decade, evolutionary and meta-heuristic algorithms have been extensively used as search and optimization tools in various problem domains, including science, commerce, and engineering. Their broad applicability, ease of use, and global perspective may be considered as the primary reason for their success. Ant colony foraging behavior may also be considered as a typical swarm-based approach to optimization. In this paper, ant colony optimization algorithm (ACO) is presented and tested with few benchmark examples. To test the performance of the algorithm, three benchmarks constrained and/or unconstrained real valued mathematical models were selected. The first example is the Ackley's function which is a continuous and multimodal test function obtained by modulating an exponential function with a cosine wave of moderate amplitude. The algorithm application resulted in the global optimal with reasonable CPU time. To show the efficiency of the algorithm in constraint handling, the model was applied to a two-variable, two constraint highly nonlinear problem. It was shown that the performance of the model is quite comparable with the results of well developed GA. The third example is a real world water resources operation optimization problem. The developed model was applied to a single reservoir with 60 periods with objective of minimizing the total square deviation from target demand. Results obtained are quit promising and compares well with the results of some other well-known heuristic approaches.

Key-Words: - Ant colony; Optimization; Mathematical problems; Reservoir operation

1 Introduction

Ant colony optimization (ACO), called ant system (Colorni et al. 1991; Dorigo 1992), was inspired by studies of the behavior of ants (Deneubourg et al. 1983). Ant algorithms were first proposed by Dorigo (1992) and Dorigo et al. (1996) as a multi-agent approach to different combinatorial optimization problems like the traveling salesman problem and the quadratic assignment problem. The ant-colony metaheuristic framework was introduced by Dorigo and Di Caro (1999), which enabled ACO to be applied to a range of combinatorial optimization problems. Dorigo et al. (2000) also reported the successful application of ACO algorithms to a number of bench-mark combinatorial optimization problems.

In this paper a ant colony optimization algorithm is developed and its performance is tested using three well defined and highly nonlinear benchmark mathematical functions, as well as developing an optimum operation policy for a single reservoir.

2 Ant colony optimization (ACO) algorithms: general aspects

Ant colony algorithms have been founded on the observation of real ant colonies. By living in colonies, ants' social behavior is directed more to the survival of the colony entity than to that of a single individual member of the colony. An interesting and significantly important behavior of ant colonies is their foraging behavior, and in particular, their ability to find the shortest route between their nest and a food source, realizing that they are almost blind. The path taken by individual ants from the nest, in search for a food source, is essentially random [4]. However, when they are traveling, ants deposit on the ground a substance called pheromone, forming a pheromone trail as an indirect communication means. By smelling the pheromone, there is a higher probability that the trail with a higher pheromone concentration will be chosen. The pheromone trail allows ants to find their way back to the food source and vice versa. The trail is used by other ants to find the location of the food source located by their nest mates. It follows that when a number of paths is available from the nest to a food source, a colony of ants may be able to exploit the pheromone trail left by the individual members of the colony to discover the shortest path from the nest to the food source and back [6]. As more ants choose a path to follow, the pheromone on the path builds up, making it more attractive to other ants seeking food and hence more likely to be followed by other ants.

In general, ACO algorithms employ a finite size of artificial agents with defined characteristics which collectively search for good quality solutions to the problem under consideration. Starting from an initial state selected according to some case-dependent criteria, each ant builds a solution which is similar to a chromosome in a genetic algorithm. While building its own solution, each ant collects information on its own performance and uses this information to modify the representation of the problem, as seen by the other ants [5]. The ant's internal states store information about the ant's past behavior, which can be employed to compute the goodness/value of the generated solution. Artificial ants are permitted to release pheromone while developing a solution or after a solution has fully been developed, or both. The amount of pheromone deposited is made proportional to the goodness of the solution an artificial ant has developed (or is developing).

Rapid drift of all the ants towards the same part of the search space is avoided by employing the stochastic component of the choice decision policy and the pheromone evaporation mechanism. To simulate pheromone evaporation, the pheromone persistence coefficient (ρ) is defined which enables greater exploration of the search space and minimizes the chance of premature convergence to suboptimal solutions (see Eq. 3). A probabilistic decision policy is also used by the ants to direct their search towards the most interesting regions of the search space. The level of stochasticity in the policy and the strength of the updates in the pheromone trail determine the balance between the exploration of new points in the state space and the exploitation of accumulated knowledge [5].

Let $\tau_{ij}(t)$ be the total pheromone deposited on path *ij* at time *t*, and $\eta_{ij}(t)$ be the heuristic value of path *ij* at time *t* according to the measure of the objective function. We define the transition probability from node *i* to node *j* at time period *t* as:

$$P_{ij}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}}{\sum_{l \in allowed}[\tau_{il}(t)]^{\alpha}[\eta_{il}(t)]^{\beta}} & \text{if } j \in \text{allowed} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where α and β = parameters that control the relative importance of the pheromone trail versus a heuristic value. Let q be a random variable uniformly distributed over [0, 1], and $q_0 \in$ [0, 1] be a tunable parameter. The next node j that ant k chooses to go is:

$$j = \begin{cases} \arg\max_{l \in allowed_k} \{[\tau_{il}(t)]^{\alpha}[\eta_{il}(t)]^{\beta}\} & if \;\; q \le q_0 \\ J & otherwise \end{cases} \quad (2)$$

where J = a random variable selected according to the probability distribution of $P_{ij}(t)$. The pheromone trail is changed globally. Upon completion of a tour by all ants in the colony, the global trail updating is done as follows:

$$\tau_{ij}(t) \xleftarrow{iteration} (1-\rho).\tau_{ij}(t) + \rho.\Delta\tau_{ij} \quad (3)$$

where $0 \le \rho \le 1$; ρ = evaporation (i.e., loss) rate; and the symbol $\xleftarrow{iteration}$ is used to show the next iteration.

There are several definitions for $\Delta\tau_{ij}(t)$ ([4], [5]). In this paper, we use three algorithms as:

1. Ant System (AS) algorithm

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{M} \tau m_{ij}^{k}(t) \quad (4)$$

$$\tau m_{ij}^{k}(t) = \begin{cases} 1/G^{k}(m) & if \quad (i,j) \in T^{k}(m) \\ 0 & if \quad (i,j) \notin T^{k}(m) \end{cases} \quad (5)$$

where $G^k(m)$ = value of the objective function for the tour $T^k(m)$ taken by the *k-th* ant at iteration *m*.

2. Ant Colony System–Iteration Best (ACSib)

$$\Delta\tau_{ij}(t) = \begin{cases} 1/G^{k_{ib}}(m) & if \quad (i,j) \in \text{tour done by ant } k_{ib} \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where $G^{k_{ib}}(m)$ = value of the objective function for the ant taken the best tour at iteration *m*.

3. Ant Colony System–Global Best (ACSgb)

$$\Delta\tau_{ij}(t) = \begin{cases} 1/G^{k_{gb}} & if \quad (i,j) \in \text{tour done by ant } k_{gb} \\ 0 & \text{otherwise} \end{cases} \quad (7)$$

where $G^{k_{gb}}$ = value of the objective function for the ant with the best performance within the past total iteration.

3 Algorithm application

To test the performance of the proposed algorithm, the model was applied to a few benchmark constrained and unconstrained mathematical optimization functions. Unconstrained optimization deals with the problem of minimizing or maximizing a function in the absence of any restrictions.

Ackley's function is a continuous and multi-modal test function obtained by modulating an exponential function with a cosine wave of moderate amplitude. Its topology is characterized by an almost flat outer region and a central hole or peak where modulations by cosine wave become more and more influential. Ackley's function is as follow:

Minimize $f(x_1, x_2) = -c_1.\exp(-c_2\sqrt{\frac{1}{2}\sum_{j=1}^{2} x_j^2}) - \exp(\frac{1}{2}\sum_{j=1}^{2}\cos(c_{3.}x_j)) + c_1 + e$ (8)

$-5 < x_j < 5 \qquad j = 1,2$ (9)

Where $c_1 = 20$, $c_2 = 0.2$, $c_3 = 2\pi$, and $e = 2.71282$.

This function causes moderate complications to the search, since though a strictly local optimization algorithm that performs hill-climbing would surely get trapped in a local optimum. A search strategy that scans a slightly bigger neighborhood would be able to cross intervening valleys toward increasingly better optima. Therefore, Ackley's function provides one of the reasonable test cases for honey bees mating search algorithm. Employing the proposed ACO algorithm, the fitness value is $f(x_1^*, x_2^*) = -.0054617$ was obtained as average of 10 runs. Using GA, at the 1000th generation, the fitness value of $f(x_1^*, x_2^*) = -.005456$ has been obtained [8]. The best rate of convergence for 10 runs is presented in Figure (1).

The second numerical example of unconstrained optimization problem is given as follows [8]:

Maximize $f(x_1, x_2) = 21.5 + x_1\sin(4\pi x_1) + x_2\sin(20\pi x_2)$ (10)

$-3.0 \le x_1 \le 12.1$ (11)

$4.1 \le x_2 \le 5.8$ (12)

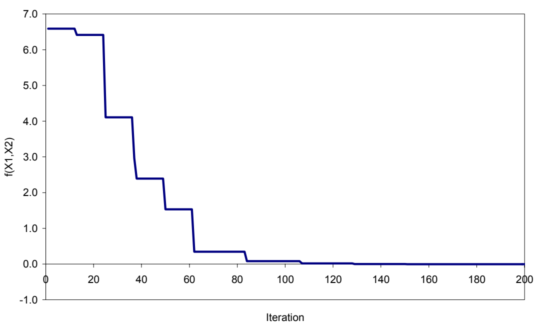

Fig. 1. *The best rate of convergence of Ackley's function value by ACO.*

Again employing the proposed ACO algorithm, the best fitness value was obtained as of 38.53283. The best rate of convergence for 10 runs is presented in Figure (2).

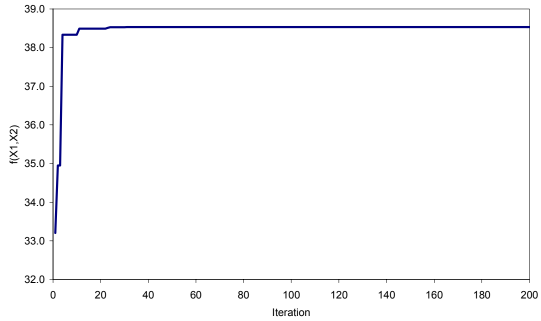

Fig. 2. *The best rate of convergence of second numerical example.*

Solving the same problem with GA, the best run was terminated after 1000 generations, obtaining the best fitness value as of 38.818208 [8].

To show the efficacy of this handling method, we apply ACO with this method to solve a two-variable, two-constraint NLP problem:

Maximize $f1(x_1, x_2) = (x_1^2 + x_2 - 11)^2 + (x_1 + x_2^2 - 7)^2$ (13)

S.T:

$g_1(x) \equiv 5.059 - x_1^2 - (x_2 - 2.5)^2 \ge 0$ (14)

$g_2(x) \equiv (x_1 - 0.05)^2 + (x_2 - 2.5)^2 - 4.84 \ge 0$ (15)

$0 \le x_1 \le 6 \quad , \quad 0 \le x_2 \le 6$ (16)

The unconstrained objective function $f1(x_1, x_2)$ has a minimum solution at (3, 2) with a function value equal to zero. However, due to the presence of constraints, this solution is no more feasible and constrained optimum solution is $x^* = (2.246826, 2.381865)$ with a function value equal to $f_1^* = 13.59085$. The feasible region is a narrow crescent-shaped region (approximately 0.7% of the total search space) with the optimum solution lying on the second constraint. Employing the same algorithm with, the best fitness value was obtained as $f(x^*, x^*) = 13.614285$.

4 ACO algorithm for optimum reservoir operation

To apply ACO algorithms to a specific problem, the following steps have to be taken: (1) Problem representation as a graph or a similar structure easily covered by ants; (2) Assigning a heuristic preference to generated solutions at each time step (i.e., selected path by the ants); (3) Defining a fitness function to be optimized; and (4) Selection of an ACO algorithm to be applied to the problem.

In optimum reservoir operation problem, links between initial and final storage volumes at different periods form a graph which represents the system, determining the release at that period.

The heuristic information on this problem is determined by considering the criterion as minimum deficit:

$$\eta_{ij}(t) = 1/([R_{ij}(t) - D(t)]^2 + c) \quad (17)$$

where $R_{ij}(t)$ = release at period t, provided the initial and final storage volume at classes i and j, respectively; $D(t)$ = demand of period t; and c = a constant to avoid irregularity (dividing by zero in Eq. 17.). To determine $R_{ij}(t)$, the continuity equation along with the following constraints, may be employed as:

$$R_{ij}(t) = S_i - S_j + I(t) - LOSS_{ij}(t) \quad (18a)$$

$$S_{\min} \le S_i \le S_{\max} \quad (18b)$$

$$S_{\min} \le S_j \le S_{\max} \quad (18c)$$

$$S_1 = S_{NT+1} \quad (18d)$$

where S_i and S_j = initial and final storage volumes (class i and j), respectively; $I(t)$ = inflow to the reservoir at time period t; $LOSS_{ij}(t)$ = loss (e.g., evaporation) at period t provided that initial and final storage at classes i and j respectively; S_{min} and S_{max} = minimum and maximum storage allowed respectively; and NT= total number of periods. Using the transition rule (Eq. 2), each ant is free to choose the class of final storage (end-of-period storage), if it is feasible through the continuity equation and storage constraints (Eqs. 18).

The fitness function is a measure of the goodness of the generated solutions according to the defined objective function. For this study, total square deviation (TSD) is defined as:

$$TSD^k = \sum_{t=1}^{NT} \left[(R^k(t) - D(t)) / D_{\max} \right]^2 \quad (19)$$

where $R^k(t)$ = release at period t recommended by ant k and D_{max} = maximum monthly demand.

The Ant Colony System–Global Best algorithm (ACSgb), have been used. The so-called solution construction and pheromone trail update rule considered by this ACO algorithm are employed.

To illustrate the performance of the model, the Dez reservoir in southern Iran, with an effective storage volume of 2,510 MCM and average annual demand of 5,900 MCM is selected. For illustration purposes, a period of 60 months with an average annual inflow of 5,303 MCM is employed. The reservoir volume is divided into 14 classes with 200 MCM intervals. To start with the model, a finite number of ants is randomly distributed in different classes of initial storage volume. It is also assumed that the starting point for ants could be any time along the 60-month operation horizon. Thus, ants are also uniformly random distributed along the operation horizon. Feasible paths for ants to follow are constrained by the continuity equation, and the minimum and maximum permitted storage volume (Eqs. 18). By completion of the first tour by all ants, there will be a finite number of feasible solutions with values for the objective function. Now, realizing the values of the fitness function, the pheromones must be updated to continue the next iteration. When the pheromone update is completed, the next iteration begins.

In this paper, after tuning the parameters of the model, the best value of the parameter choose as follow:

Number of ants=100; Number of iterations=500; $\tau_0 = 1$; ρ= 0.25, α = 1, β = 4, and q_0 = 0.9.

The best overall result obtained from ACSgb for initial and final storage volumes of 1,430 MCM is 1.296 (TSD). The global optimum with the same initial and final storage volumes resulted in TSD = 1.273. Clearly, the developed model with the ACSgb algorithm for pheromone updating provides comparable results with those of global optimum, and seems promising in optimum reservoir operation. The fluctuation of reservoir release, taken from two models is presented in Fig. 3. Except for a few months, reservoir releases resulting from the proposed algorithm follow those of global optimum very well.

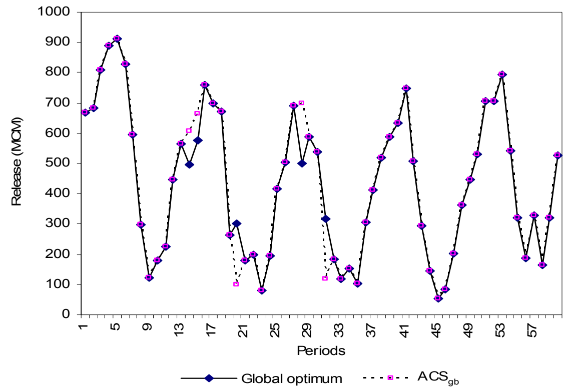

Fig. 3. *Comparison of reservoir releases resulting from ACSgb and global optimum.*

5 Concluding remarks

While walking from one point to another, ants deposit a substance called pheromone, forming a pheromone trail. It has been shown experimentally that this pheromone trail, once employed by a colony of ants, can give rise to the emergence of a shortest path. In general, the amount of pheromone deposited is made proportional to the goodness of the solution an ant may build.

Modeling ant colony behavior as an optimization algorithm and its application to few benchmarks, highly nonlinear-constrained and/or unconstrained optimization problems, such as well known Ackley's function, partially revels the high potential of the proposed algorithm to solve nonlinear optimization problems. Results obtained are well comparable with these obtained employing well developed GAs, is promising.

The model performance in real world water management problems, such as reservoir operation, proved to be very promising. The problem may be approached by considering a time series of inflow, classifying the reservoir volume to several intervals, and deciding on the release at each period with respect to an optimality criterion. Feasible paths for ants to follow may be constrained by the continuity equation as well as constraints on the storage volume. Upon each tour completion, a finite number of feasible solutions will form, leaving a new value for the pheromone.

Realizing the values of the fitness function, the pheromones will be updated by global and local update rules. Application of the proposed model to the Dez reservoir in Iran provided promising results. As for any search method, the performance of the proposed model is quite sensitive to setup parameters, hence fine tuning of the parameters is recommended.

References:

[1] Colorni, A., Dorigo, M., and Maniezzo, V., "Distributed optimization by ant-colonies" *Proc., 1st European Conf. on Artificial Life (ECAL'91),* Cambridge, Mass, USA, MIT Press, 1991, pp. 134-142.

[2] Deneubourg, J. L., Pasteels, J. M., Verhaeghe, J. C., "Probabilistic behavior in ants: A strategy of errors?" *J. Theoretical Biology*, 105, 1983, pp. 259-271.

[3] Dorigo, M., "Optimization, learning and natural algorithms" Ph.D. Thesis, Politecnico di Milano, Italy, 1992.

[4] Dorigo, M., Maniezzo, V., and Colorni, A., "The ant system: optimization by a colony of cooperating ants." *IEEE Trans. Syst. Man. Cybern*., 26, 1996, pp. 29-42.

[5] Dorigo, M., and Gambardella, L. M., "Ant colony system: A cooperative learning approach to the traveling salesman problem." *IEEE Transactions on Evolutionary Computation,* 1(1), 1997, pp. 53-66.

[6] Dorigo, M., and Di Caro, G., "The ant colony optimization metaheuristic." *New ideas in optimization*, D. Corne, M. Dorigo, and F. Glover, eds., McGraw-Hill, London, 1999, pp. 11-32.

[7] Dorigo, M., Bonabeau, E., and Theraulaz, G., "Ant algorithms and stigmergy." *Future Generation Comput. Systems*, 16, 2000, pp. 851-871.

[8] Gen, M. and Cheng, R. W., *Genetic Algorithms and Engineering Design*. John Wiley & Sons, Inc, 1997

Notations:

The following symbols are used in this paper:

ρ : pheromone evaporation.

$P_{ij}(t)$: transition probability from node *i* to node *j* at time period *t*.

$\tau_{ij}(t)$: total pheromone deposited on path *ij* at time *t*.

$\eta_{ij}(t)$: the heuristic value of path *ij* at time *t*.

α , β : parameters that control the relative importance of the pheromone trail versus a heuristic value.

q : a random variable uniformly distributed over [0, 1].

q_0 : be a tunable parameter $\in$ [0, 1].

τ_0 : initial value of pheromone.

$\Delta\tau_{ij}(t)$: total change in pheromone of path *ij* at time period *t*.

$\tau m_{ij}^{k}(t)$: change in pheromone of path *ij* at time period *t* associated to ant *k*.

$G^k(m)$: value of the objective function of ant *k* at iteration *m*.

$T^k(m)$: the tour taken by ant *k* at iteration *m*.

$G^{k_{ib}}(m)$: value of the objective function for the ant taken the best tour at iteration m.

$G^{k_{gb}}$: value of the objective function for the ant with the best performance within the past total iteration.

$R_{ij}(t)$: release at period *t*.

$D(t)$: demand of period *t*.

c : a constant.

S : storage.

$I(t)$: inflow to the reservoir at time period t.

$LOSS_{ij}(t)$: loss (e.g., evaporation) at period *t* provided that initial and final storage at classes *i* and *j* respectively.

S_{min} : minimum storage allowed.

S_{max} : maximum storage allowed.

NT: total number of periods.

TSD: total square deviation.

$R^k(t)$: release at period t recommended by ant k.

D_{max} : maximum monthly demand.