Water Resources Management (2006) 20: 879–898
DOI: 10.1007/s11269-005-9012-0

Ant Colony Optimization for Multi-Purpose Reservoir Operation

D. NAGESH KUMAR* and M. JANGA REDDY
Department of Civil Engineering, Indian Institute of Science, Bangalore-560 012, India
*(*author for correspondence, e-mail: nagesh@civil.iisc.ernet.in)*

(Received: 10 May 2005; in final form: 28 November 2005)

Abstract. In this paper a metaheuristic technique called Ant Colony Optimization (ACO) is proposed to derive operating policies for a multi-purpose reservoir system. Most of the real world problems often involve non-linear optimization in their solution with high dimensionality and large number of equality and inequality constraints. Often the conventional techniques fail to yield global optimal solutions. The recently proposed evolutionary algorithms are also facing problems, while solving large-scale problems. In this study, it is intended to test the usefulness of ACO in solving such type of problems. To formulate the ACO model for reservoir operation, the problem is approached by considering a finite time series of inflows, classifying the reservoir volume into several class intervals, and determining the reservoir release for each period with respect to a predefined optimality criterion. The ACO technique is applied to a case study of Hirakud reservoir, which is a multi-purpose reservoir system located in India. The multiple objectives comprise of minimizing flood risks, minimizing irrigation deficits and maximizing hydropower production in that order of priority. The developed model is applied for monthly operation, and consists of two models viz., for short-time horizon operation and for long-time horizon operation. To evaluate the performance of ACO, the developed models are also solved using real coded Genetic Algorithm (GA). The results of the two models indicate that ACO model performs better, in terms of higher annual power production, while satisfying irrigation demands and flood control restrictions, compared to those obtained by GA. Finally it is found that ACO model outperforms GA model, especially in the case of long-time horizon reservoir operation.

Key words: Ant Colony Optimization, Genetic Algorithms, reservoir operation, hydropower, irrigation, flood control

1. Introduction

Most of the real world optimization problems often involve large scale optimization. In the past, many optimization techniques used to find optimal solutions were constrained by the complexities of non-linear relationships in model formulation and by increase in the number of variables and constraints. For this reason, recently many heuristic and metaheuristic algorithms have been proposed, which though do not always ensure the global optimum solution, however give quite good results in an acceptable computation time. So researchers are persistently looking for newer techniques and their improvements over the years. Recently, a new metaheuristic technique, namely Ant Colony Optimization (ACO) technique has been proposed and is becoming increasingly popular in tackling various large-scale optimization problems (Dorigo *et al.*, 1996). In the field of water resources, reservoir operation is one such problem that involves many complexities in its operation. In this paper, the potential of ACO technique to derive suitable operating policies for a multi-purpose reservoir system is presented.

1.1. SCOPE FOR NON LINEAR OPTIMIZATION IN RESERVOIR OPERATION

In the case of multi-purpose reservoir operation, the goals are much more complex than for single purpose reservoir operation and often involve several problems like insufficient inflows and higher demands. In order to satisfy the objectives to the maximum extent with the available water resources, in the face of inherent constraints, it is necessary to operate the reservoir system in the best possible way, to get maximum benefits with minimum losses. Optimally operating a reservoir is a complicated task and there is no standard algorithm available which can be applicable to all problems, since each problem contains its unique physical and operating characteristics (Yeh, 1985).

In the past, many traditional optimization algorithms have been used for management of complex water resources systems in order to provide an improved basis for decision making. While formulating the model to solve a problem closer to reality, it often leads to nonlinearities and nonconvexities in their objectives and constraints. So the conventional techniques have difficulties in solving real world problems. For example, a typical hydropower production is an intricate function of non-linear objectives and constraints. Though generalized softwares are available for solving linear programming (LP) problems, the strictly linear form of LP does limit its applicability (Wurbs, 1993). Similarly, with increase in the number of state variables, the dynamic programming solutions face the problem of curse of dimensionality, i.e., any increase in number of state variables cause exponential increase in computation time (Yeh, 1985). The conventional non-linear programming techniques have the problem of getting trapped to local optima and also computational requirements are huge (Yeh, 1985).

To overcome those problems, in recent times, Evolutionary Computation (EC) techniques have been proposed and applied to various kinds of problems in water resources field. Genetic Algorithm (GA) is one such EC technique. The GA is basically a Darwinian natural selection process, that combines the concept of survival of the fittest with natural genetic operators (Holland, 1975). The working of GAs and its application are well documented in Goldberg (1989) and Michaelwicz (1996). In the field of water resources, for reservoir operation, few applications of GA technique to derive reservoir-operating policies have been reported recently (Oliveira and Loucks, 1997; Wardlaw and Sharif, 1999; Chang and Chen, 1998; Sharif and Wardlaw, 2000; Nagesh Kumar *et al.*, 2005). In GA the increased dimensionality due to high number of decision variables and constraints, does not impose greater computational cost, but the number of function evaluations required to get at an optimum is very large. Also with very long chromosomes, the likelihood of obtaining the global optimum is considerably reduced. Recently researchers observed that metaheuristic technique like Ant Colony Optimization (ACO) is making some improvement in this direction for discrete combinatorial optimization (Dorigo *et al.*, 1996). Few researchers found that ACO outperforms other evolutionary optimization algorithms including GAs (Dorigo and Gambardella, 1997; Maier *et al.*, 2003). It was also inferred that as the search space becomes larger the performance of ACO improves significantly over GAs (Dorigo *et al.*, 1999).

From literature, it is observed that ACO applications to water resources problems are quite few and there is a great potential to apply ACO in this field. Abbaspour *et al.* (2001) employed ACO algorithm to estimate hydraulic parameters of unsaturated soil and concluded that ACO is able to estimate the true parameters within a reasonable accuracy. Maier *et al.* (2003) used ACO algorithms to find a near global optimal solution to a water distribution system, illustrating that ACO algorithm may form an attractive alternative to GAs for the design of optimum water distribution systems. Jalali *et al.* (2003) proposed ACO algorithms for monthly operation of reservoir system. In their study three alternative formulations of ACO algorithms were tested for a single purpose reservoir operation. But they have not explored the potential of ACO for large scale optimization problems. So in this study, efforts are made to explore the potential of ACO application to solve higher dimensional and highly constrained non-linear optimization problem, with multiple purposes.

In this paper two models are analysed. First one is short-time horizon operation (12 time periods at a time) with a smaller number of variables and constraints, hereafter referred as Reservoir System Operation Model - I (RSOM-I). The second model is longtime horizon operation (432 time periods at a time, which is considered to be long enough to assume stationarity of inflow data) with larger number of decision variables and constraints, hereafter referred as Reservoir System Operation Model - II (RSOM-II). The performance of ACO model is evaluated by comparing with GA model results, with respect to their efficiency in yielding an optimal solution. Therefore, the objectives of the present study include:

1. Developing a methodology for ACO to apply for a reservoir operation problem, and exploring the potential of ACO by applying it to an existing multi-purpose reservoir system.
2. Evaluate the performance of the ACO, by comparing with the results of GA.

In the following sections, a brief description about ACO and its procedure is presented first. Next, the details of the case study and model formulation for reservoir operation are explained. Later, ACO application for the reservoir operation is presented. Finally the results are discussed, followed by the conclusion.

2. Ant Colony Optimization

Ant Colony Optimization (ACO) is a metaheuristic approach proposed by Dorigo (1992). The inspiring source of ACO is the foraging behavior of ants. This behavior enables ants to find shortest paths between food source and their nest (Dorigo *et al.*, 1996; Dorigo and Stützle, 2004). While walking from their nest to food source and back, ants deposit a substance called pheromone on the ground. When they decide to go, they choose with higher probability paths that are marked by stronger pheromone concentrations. This basic behavior is the basis for a cooperative interaction and emergence of shortest paths.

Ant colony algorithms provide a multi-member approach, for solving discrete combinatorial optimization problems. ACO takes elements from real ant behavior to solve more complex problems than those faced by real ants. The first ACO algorithm presented in the literature is called Ant System (AS) (Dorigo *et al.*, 1996). Later many developments have taken place in ACO methodologies. An improvement of AS is Ant Colony System (ACS) algorithm proposed in Dorigo and Gambardella (1997). Pseudo-code for ACS algorithm is given in Figure 1. More recently Stutzle and Hoos (2000) introduced Max-Min Ant system (MMAS) algorithm and validated the same by applying to few test cases in Traveling Salesman problem (TSP) and Quadratic Assignment Problem (QAP) and concluded that MMAS was one of the best in performance. A brief description of ACO algorithm based on Dorigo and Gambardella (1997) is presented in the next section.

2.1. ACO ALGORITHM

In ACO, each ant builds a possible solution to the problem, by moving through a finite sequence of neighbor states. Moves are selected by applying a stochastic local

```
Procedure of ACS Algorithm:
Begin
    Initialize
    While stopping criterion not satisfied do
        Position each ant in a starting node
        Repeat
            For each ant do
                Choose next node by applying the state transition rule
                Apply step by step pheromone update
            End for
        Until every ant has built a solution
        Update best solution
        Apply offline pheromone update
    End While
End
```

Figure 1. Pseudo-code for the Ant Colony System (ACS) algorithm.

search directed by the ant internal state, problem specific heuristic information and the shared information about the pheromone.

Ants use a decision rule called *pseudo-random proportional rule*, in which, an ant *k* in node *i* will select node *j* to move as follows

$$P_{ij}^{k}(t) = \begin{cases} \dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}}{\sum_{u \in J^{k}(i)}\{[\tau_{iu}(t)]^{\alpha}[\eta_{iu}(t)]^{\beta}\}} & \text{if } j \in J^{k}(i) \\ 0 & \text{if } j \notin J^{k}(i) \end{cases} \tag{1}$$

where, $\eta_{ij}(t)$ represents heuristic information about the problem i.e., the heuristic value of path *ij* at time *t* according to the measure of the objective function; $\tau_{ij}(t)$ represents the total pheromone deposited on path *ij* at time t; $J^{k}(i)$ represents the allowable moves for ant *k* from node i; α and β are parameters that determine the relative importance of the pheromone trail with respect to the heuristic information.

The *state transition rule* is as follows: the next node *j* that ant *k* chooses to go is given as,

$$J = \begin{cases} \max\limits_{u \in J_{k}(i)} \left\{[\tau_{iu}(t)]^{\alpha}[\eta_{iu}(t)]^{\beta}\right\} & \text{if q} \le \text{q}_0 \\ J & \text{if q} > \text{q}_0 \end{cases} \tag{2}$$

where q is a random number uniformly distributed in [0, 1]; q_0 is a tunable parameter $(0 \le \text{q}_0 \le 1)$; $J \in J^{k}(i)$ is a node randomly selected according to the probability distribution given by Equation 1.

In each iteration of the algorithm, each ant progressively builds a solution, by using the probability transition rule. The pheromone trail is updated both locally and globally.

Local updating: During the construction of the solution, if an ant carries out the transition from node *i* to node *j*, then the pheromone value of the corresponding arc will be changed as,

$$\tau_{ij}(t) \xleftarrow{step} (1-\varphi).\tau_{ij}(t) + \varphi.\tau_0 \tag{3}$$

where τ_0 is the initial value of pheromone; φ is a tunable parameter $(0 \le \varphi \le 1)$. Local updating is very useful to avoid premature convergence of the solution, and helps in exploring the new search space, for the problems where the starting node is fixed.

Global updating: At the end of an iteration of the algorithm, once all the ants have built a solution each, pheromone trail is added to the arcs used by the ant that found the best tour from the beginning of the trail. The global trail updating rule is given as

$$\tau_{ij}(t) \xleftarrow{iteration} (1-\rho).\tau_{ij}(t) + \rho.\Delta\tau_{ij} \tag{4}$$

where $\rho \in [0, 1]$ is a persistence parameter that controls the pheromone decay; $\Delta\tau_{ij}$ is increment in pheromone trail and according to ACS (Dorigo and Gambardella, 1997) is given as,

$$\Delta\tau_{ij}(t) = \begin{cases} \frac{1}{F_{gb}} & \text{if } (i, j) \in \text{global best tour} \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where, F_{gb} is a fitness function corresponding to the global best tour within all the past iterations. Sometimes another type of global best ant updating rule called iteration best ant update is used. The updating rule is the same as given in Equation 4, but in Equation 5, F_{gb} will be replaced by F_{ib}, the fitness function corresponding to the best tour done by any ant in the current iteration.

It can be noted that ACO is a problem dependent application. So to apply the algorithm, it requires appropriate representation of the problem and suitable heuristics in its solution construction (Dorigo and Di Caro, 1999).

3. Case Study Description

The case study considered in this paper is the Hirakud reservoir project in Orissa state, India. Hirakud dam is situated at latitude 21°32′ N and longitude 83°52′ E. The index map of Mahanadi river basin showing the location of Hirakud dam is presented in Figure 2. The reservoir has an active storage capacity of 5,375 Mm^3 (Million cubic meters) and a gross storage of 7,189 Mm^3. The Hirakud project is a multi-purpose scheme and the water available in the dam is used in the following order of priority: for flood control, drinking water, irrigation, and power generation. Since the drinking water requirement is a very small quantity compared to other demands, this quantity is neglected in this particular model formulation. Water levels begin rising in July with the beginning of monsoon season in the region, and begin declining in October, at the end of the season. During monsoon season, the project provides flood protection to 9,500 km^2 of delta area in the districts of Cuttack and Puri. Also the project provides irrigation for 155,635 ha in wet season (Kharif) and for 108,385 ha in dry (Rabi) season in the districts of Sambalpur, Bargarh, Bolangir, and Subarnpur. The water released through the powerhouses after power generation, irrigates further 436,000 ha of command area in Mahanadi delta. Installed capacity of power generation is 198 MW through its two powerhouses at Burla (PH-I) located at the right bank and Chiplima (PH-II) located at 22 km downstream of the dam (Proc. of 3rd meeting of rule curve revision committee, 1988). The PH-I generates energy by utilizing water discharged directly from the Hirakud dam. Then the utilized water passes to the PH-II through a power channel to generate further power at Chiplima.

Orissa state is having plenty of water during the wet season, so there is great possibility for hydropower improvement in that season. Net energy production is high during the monsoon period. However, unless the region experiences unusually

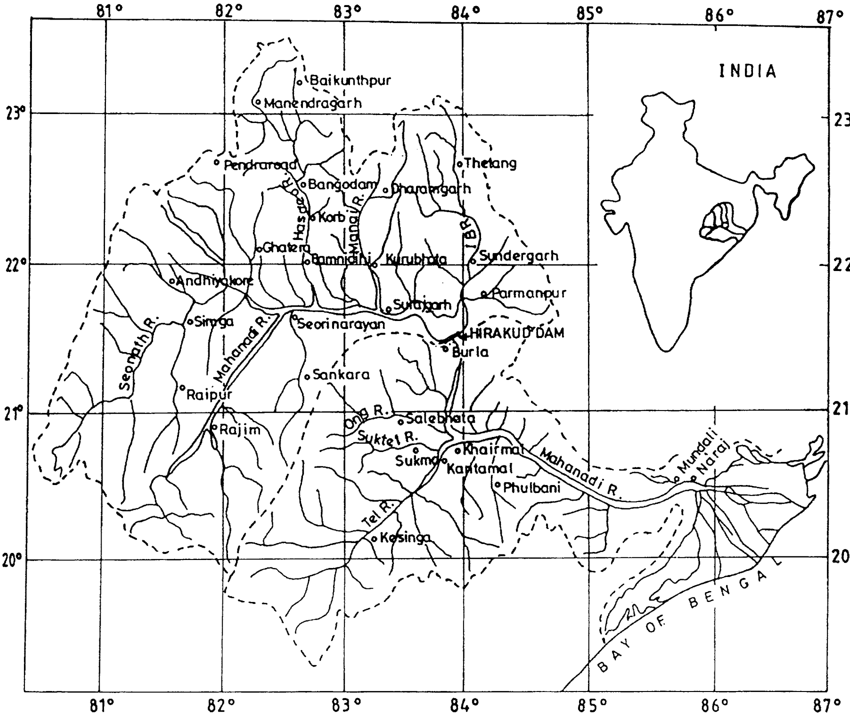

Figure 2. Index map of Mahanadi river basin shows location of Hirakud dam.

heavy rain in the dry season, power generation would not be possible in that season. Otherwise power production will be little during dry season. Over a period of 36 years the average annual inflow is 3.36 Mha-m (Million hectare meters). The reservoir inflow, utilization pattern and dam details were collected from Department of Irrigation, Government of Orissa. The historic inflow data was available for 36 years from 1958 to 1993. Monthly data of 432 periods for those 36 years is used in this study.

3.1. RESERVOIR OPERATION MODEL

To handle multiple objectives of the reservoir system, the constraint approach is adopted in this study to convert the multiple objectives problem into a single objective problem. The objective function of the model is maximizing hydropower production subject to flood rule curve restrictions, irrigation release constraints and other physical and technical constraints. The model is formulated for monthly operation, as follows:

Minimize sum of squared deviation of monthly targeted power,

$$\text{Max } E = \sum_{t=1}^{NT} (P_t - TP_t)^2 \tag{6}$$

where, P_t is hydropower produced in MkWh during period t and TP_t is targeted power in MkWh for period t; and NT = total number of time periods.

Hydropower (P_t) produced is given by

$$P_t = k_1 * RP(t) * H(t) \tag{7}$$

where, k_1 is power coefficient; $RP(t)$ is the amount of water released to turbines during period 't'; $H(t)$ is the average head available during period 't' and is expressed as a non-linear function of the average storage during that period;

Subject to the following constraints:

Mass balance equation for the reservoir storages and inflows is

$$S(t+1) = S(t) + I(t) - RP(t) - IRR(t) - E(t) - OF(t) \qquad \textit{for all } t = 1, 2, \ldots, NT \tag{8}$$

where, $S(t)$ and $S(t+1)$ are initial and final storage volumes respectively during time period t; $I(t)$ is inflow into the reservoir; $IRR(t)$ is irrigation release; $E(t)$ is the evaporation losses and is expressed as a non-linear function of the average storage during that period; $OF(t)$ is the overflow or spill from the reservoir.

Storage bounds for the reservoir:

$$S_{\min}(t) \leq S_i(t) \leq S_{\max}(t) \qquad \textit{for all } t = 1, 2, \ldots, NT \tag{9}$$

where, $S_{\min}(t)$ and $S_{\max}(t)$ are minimum and maximum storages allowed in period t respectively. These storages are constrained by flood protection rules during monsoon season. In the remaining periods, minimum storage ($S_{\min}(t)$) is taken as equal to dead storage, while maximum storage ($S_{\max}(t)$) is equivalent to the full capacity of the reservoir.

Irrigation release constraints:

$$IDEM_{\min}(t) \leq IRR(t) \leq IDEM_{\max}(t) \qquad \textit{for all } t = 1, 2, \ldots, NT \tag{10}$$

where, $IDEM_{\min}(t)$ and $IDEM_{\max}(t)$ are minimum and maximum irrigation demands respectively in time period t;

Turbine capacity constraint:

$$0 \leq RP(t) \leq TC \qquad \textit{for all } t = 1, 2, \ldots, NT \tag{11}$$

where, TC is the turbine capacity of power plant.

In addition to the above constraints, it is to be ensured that end storage of the last period is equal to the initial storage of the first period.

$$S_1 = S_{NT+1} \tag{12}$$

In this model, the first priority is given for flood control, and the next priority to meet irrigation demand, and then to hydropower demand. If any excess water is found after meeting these two demands, such water will be spilled out as overflow, for utilization by diversion structures downstream.

4. Optimal Reservoir Operation using Ant Colony Optimization

As mentioned earlier ACO is a problem dependent application. So to apply ACO algorithm, the following steps need to be considered:

1. An appropriate representation of the problem, as a graph or a similar structure easily covered by ants, which facilitates the incremental construction of possible solutions, using a probabilistic transition rule to move from one state i to a neighboring state j.
2. Selection of heuristic information $\eta_{ij}(t)$, that provides the problem specific knowledge to be used by the search process to move from node i to node j.
3. Defining an appropriate fitness function to be optimized for the problem.
4. Selection of proper pheromone updating rules, which best suit for the given problem.

4.1. REPRESENTATION OF THE PROBLEM

To formulate the ACO model for reservoir operation, it is expedient to consider the problem as a combinatorial optimization problem, with a facility of graphical representation. The reservoir volume is divided into several classes for each time period to make a combinatorial optimization problem. A typical graphical representation of the solution approach is shown in Figure 3. The problem is approached by considering a time series of inflow, classifying the reservoir volume into several intervals, and deciding the releases for each period with respect to a predefined optimality criterion. Links between initial and final storage volumes of different periods form a graph, which represents the system, determining the release during that period. Figure 4 explains the initial distribution of ants over different time periods, in different storage classes at the beginning of iteration.

4.2. HEURISTIC INFORMATION

The heuristic information $\eta_{ij}(t)$ about the problem is determined by including the minimum squared deviation in the criterion.

$$\eta_{ij}(t) = \frac{1}{\left[\frac{RP(t) - PD(t)}{PD(t)}\right]^2 + c} \quad \text{for all } t = 1, 2, \ldots, NT \tag{13}$$

Where, $RP(t)$ = water release for power made in period t, with the initial and final storage volumes at classes i and j, respectively; $PD(t)$ = water release to be made

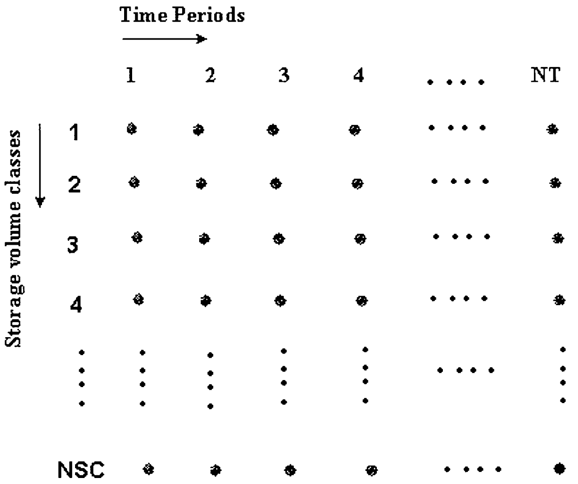

Figure 3. Graphical representation of reservoir operation problem, showing the discretization of storage volume into several classes for each time period. NT = total number of time periods; NSC = total number of storage classes.

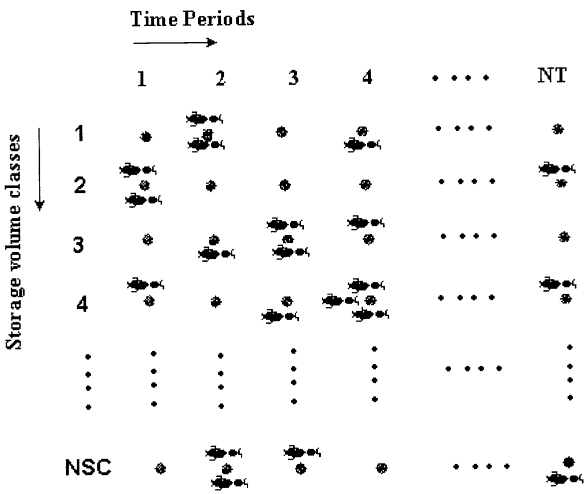

Figure 4. Typical layout showing random distribution of ants along the time horizon and storage volume classes at the start of iteration.

to generate targeted power in period t; and c = a constant to avoid divisibility by zero.

4.3. FITNESS FUNCTION

The fitness function is a measure of the goodness of the generated solutions according to the defined objective function. For this study, minimum total squared deviation (TSD) is used as a fitness measure. To limit the range of values, the fitness function is normalized. The fitness function is given as:

$$TSD = \sum_{t=1}^{NT} \left[\frac{(RP(t) - PD(t))}{PD(t)} \right]^2 \tag{14}$$

It may be noted that the model formulation consists of various equality and inequality constraints (Equations 6 to 12) as described in the previous section and handling of these constraints is explained in section 5.

4.4. PHEROMONE UPDATING RULES

The effect of local and global updating rules is tested individually for the reservoir operation model. After a number of trials, it is found that without local updating, the algorithm results in better performance. So in this study only global updating is used. In the case of global updating, when using only global best ant (the ant corresponding to best fitness value from the start of iterations) for pheromone updating, the search may concentrate too fast around this solution, and exploration of the new search space is limited, which may lead to poor quality solutions (or local optima). To avoid this, along with global best ant updating, current iteration best ant updating is chosen for the pheromone trail update. So the mixed strategy of iteration best tour trail updating for pheromone at the end of each iteration, and global best tour tail updating at regular intervals of iterations is employed in this study.

5. Model Application

As already stated above, to demonstrate the performance of the model, the Hirakud reservoir system is taken up as the case study. The reservoir serves for multiple purposes of flood control, irrigation and power production, in that order of priority. From the inflow data series, it is observed that monsoon season July, August, September (JAS) is the peak period for the reservoir to get filled and at the same time, there is imminent danger of flood occurring in these three months. So it is essential to incorporate flood protection measures into the model. The minimum and maximum levels to be maintained in JAS months are shown in Table I (Proc. of rule curve revision committee, 1988). These levels are maintained in the reservoir by putting them as constraints into the model, in the form of storage bounds, to ensure that the order of priority of usages is maintained as per the reservoir management and regulation policies. These storage bounds are the values corresponding to the levels to be maintained as per the flood protection rule curves. Irrigation demands are also considered as constraints in the model. After satisfying these constraints, the focus is then on maximizing power production with the available water. Active storage capacity of the reservoir is 5,375 Mm^3. The reservoir volume is divided into 200 classes with 26.875 Mm^3 intervals, which is much higher in number when

Table I. Reservoir storage levels to be maintained as per flood protection rules (Proc. of 3rd meeting of rule curve revision committee, 1988)

Month and Date	Min. level (m)	Max. level (m)
July 1st	-	181.356
August 1st	-	179.832
September 1st	188.928	191.1096

compared to the number of discretisation levels used in conventional optimization techniques like dynamic programming.

To initiate the model, a finite number of ants are randomly distributed into different classes of initial storage volume. It is assumed that the starting point for individual ants could be at any time period along the NT-month operation horizon. Thus, ants are also randomly distributed along the operation horizon (Figure 4). In construction of the solution, feasible paths to be followed by the ants are constrained by the continuity equation and other constraints, as shown in Equations 8 to 12. Here a feasible solution produced by an ant means, it covers the whole time period of length NT, duly satisfying the various constraints of the model. Also the solution should include exactly 1 edge from vertical nodes in Figure 4. In case the ant starts from an edge different from time period 1, then from that time period, it moves forward step-by-step one period at a time until it reaches the last time period NT. Then it goes to the time period 1, duly satisfying continuity and the initial storage constraints. From this step onwards, it moves step-by-step forward until it reaches its starting node with the same initial storage value. This is a cyclic procedure.

After completion of the tour by all ants, by taking the values of the fitness function, the pheromones are updated using global updating rules to continue with the next iteration. In the mixed strategy pheromone updating, iteration best ant trail updating is done at the end of each iteration, and global best ant trail updating is done at regular intervals of once in five iterations. When the pheromone update is completed, the next iteration is begun. The step-by-step procedure for ACO algorithm for the reservoir operation is presented in Figure 5. To handle the constraints, simulation and evaluation approach is used. At each generation, the decision variables are evaluated for the current solution and then the bounds are checked. If there is any violation in satisfying the constraints, then penalty can be applied, by choosing suitable penalty coefficient. To avoid any disruption that may be caused in convergence by using a solely penalty function approach, heuristics are used in the component selection process, which automatically satisfy any number of practical or construction constraints without using a penalty function.

To compare the performance of ACO, another naturally inspired global optimization method GA is used. GA model consist of the same objective function

1. Initialization:

   Discretize the storage volume into several classes for each time period

   Initialize pheromone trail τ_{ij}, and other parameters (e.g., α, β, ρ values)

2. Solution construction:

   Position each ant in a starting node (i.e., Random distribute the starting period and class of initial reservoir volume for each ant)

   For each ant k **do**

   Repeat

   Compute heuristic information η_{ij}

   Choose next node by applying the state transition rule given by equations (1) & (2)

   Until every ant has built a solution

   Compute fitness value given by equation (14)

   End for

3. Trail update:

   Update the best solution

   Apply offline pheromone trail update

   For iteration best ant /global best ant move (i j) **do**

   Compute $\Delta\tau_{ij}$

   Update the trail value by means of equations (4)

   End for

4. Terminating condition:

   If (end condition = = true)

   then print the best result so far

   else go to step 2

Figure 5. Step by step procedure of ACO algorithm for Reservoir operation.

and constraints as discussed in the reservoir operation model. A real coded GA is applied to the model, after carrying out a thorough sensitivity analysis for choosing its parameters. Similar to the above constraint handling approach, self adaptive and penalty function approach is used to handle the constraints of the problem.

The formulated ACO and GA models are implemented in C language and the developed programs are executed on a 1.4 GHz, 512 MB RAM, Pentium 4 PC. The developed models are applied for two different time horizons of reservoir operation. In RSOM-I the reservoir operation is for one year, in which NT = 12. In RSOM-II, the reservoir operation is for a sufficiently long period, about 36 years (NT = 432), which can well represent the stationarity in the time series. The results of these two models are discussed in the next section.

6. Results and Discussion

6.1. RESERVOIR SYSTEM OPERATION MODEL-I (RSOM-I)

To apply the model, the values of parameters of ACO should be decided. After preliminary analysis and suggestions from earlier studies on ACO, the parameters were adopted as $\alpha = 1$, $\beta = 4$, $q_0 = 0.9$, $\rho = 0.1$ for conducting sensitivity analysis. The value of the parameter, which is of interest in the sensitivity, is progressively changed and sensitivity analysis is carried out. In literature various researchers have reported a range of values for α and β based on their problem of interest. For example, Dorigo *et al.* (1996) recommended $\alpha = 1$ and $\beta = 5$ for TSP; Dorigo and Gambardella (1997) suggested $\alpha = 0.1$ and $\beta = 2$ for TSP and Zecchin *et al.* (2005) suggested $\alpha = 1$ and $\beta = 0.5$ for water distribution system problem. Based on preliminary analysis, $0 \leq \alpha \leq 4; 0 \leq \beta \leq 5$ are tested in this study for model performance. It is found that best performance of the algorithm occurs at $\alpha = 1$ and $\beta = 2$. Sensitivity analysis for different values of ρ is carried out and it is found that best performance of the algorithm occurs at $\rho = 0.1$. To study the effect of random proportion rule on the performance of the algorithm, sensitivity analysis is carried out by changing the values of q_0 and it is found that the value of $q_0 = 0.8$ is yielding better performance of the model. Similarly after testing for different combinations of number of ants (m) and maximum number of iterations ($I_{\max}$), the values chosen are $m = 10$ and $I_{\max} = 50$.

Similarly to apply GA model, sensitivity analysis for GA parameters is carried out. Since the problem involves real parameter variables, the model uses simulated binary crossover and polynomial mutation operators (Deb, 2000). The best parameters for genetic operators are found to be crossover probability (p_c) = 0.8; mutation probability (p_m) = 0.09; the distribution index for simulated binary crossover is 10 and that for mutation operator is 100; population size = 200 and generation size = 500.

For RSOM-I, the total number of decision variables is 12 and total number of constraints is 73. The developed model is run individually for all the 36 years. It can be noted that, the result reported for each year is the best result of 10 independent runs. The results of RSOM-I are shown in Figures 6 to 8. From Figure 6 it can be observed that the annual hydropower production obtained from ACO is well matching with that from GA. However ACO is yielding in marginally higher average hydropower production about 1511.83 Mkwh (Million kilowatt hours), as compared to 1502.279 Mkwh by GAs. The average monthly hydropower productions obtained using ACO and GA are shown in Figure 7. It can be observed that, for more number of time periods, ACO is performing better than GA model. Figure 8 shows the average reservoir storages at the beginning of each month. It is observed that the average computational time required for 10 independent runs, for ACO and GA are 7.84 sec and 8.09 sec respectively.

After achieving satisfactory performance for short-time horizon operation, the study shifted to deriving reservoir-operating policies for longer-time horizon i.e., over a stretch of 432 (36×12) periods at a time.

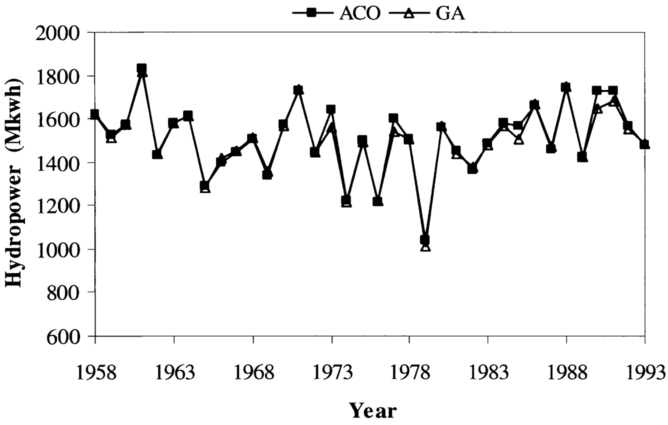

Figure 6. Annual hydropower production obtained using ACO and GA for RSOM-I.

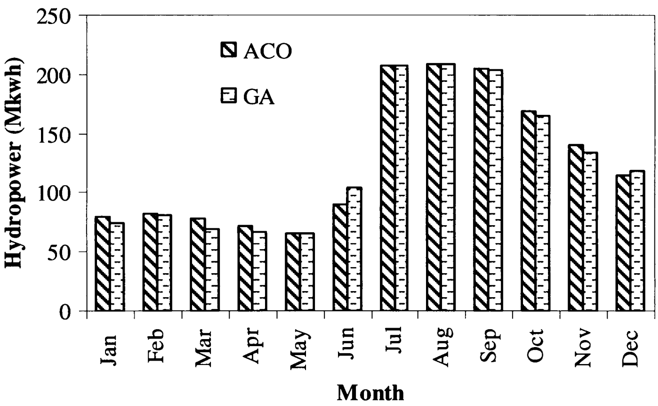

Figure 7. Monthly average hydropower production obtained using ACO and GA for RSOM-I.

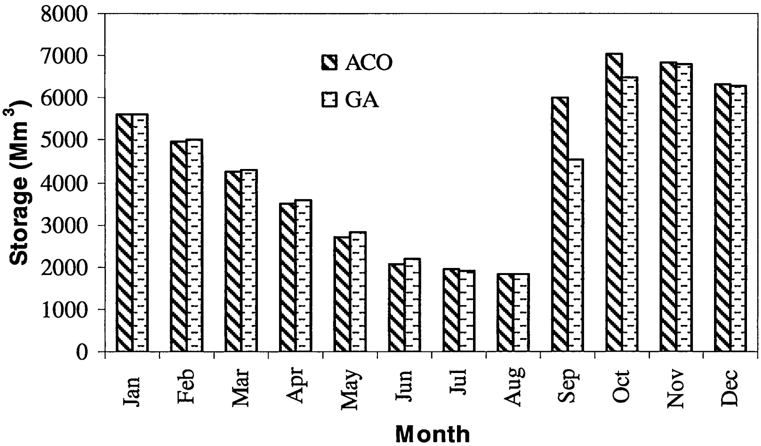

Figure 8. Monthly average initial storages obtained using ACO and GA for RSOM-I.

6.2. RESERVOIR SYSTEM OPERATION MODEL - II (RSOM-II)

Similar to RSOM-I, a thorough sensitivity analysis is carried out for ACO for different parameter settings, the best parameters found and adopted in this model are: $\alpha = 1, \beta = 2, q_0 = 0.8, \rho = 0.1$, m = 20 and $I_{max} = 200$. In the case of GA model, the parameters adopted are: crossover probability = 0.8 and mutation probability = 0.02. The distribution index for simulated binary crossover is 10 and that for mutation operator is 100. The other parameters adopted are, population size = 500 and maximum number of generations = 5,000. It may be noted that for both ACO and GA, the results reported are based on the best result obtained from 10 independent runs.

Figure 9 shows the results of annual hydropower production obtained using ACO and GA techniques. It can be seen that AGO model is resulting in higher annual hydropower production and out-performing GA. Average power production obtained by ACO is 1,533.679 Mkwh, whereas that obtained using GA is only 966.636 Mkwh. This may be due to the fact that, GA is unable to cope up with too many decision variables and unable to satisfy the large number of constraints. This model in total has 432 decision variables and 2593 constraints in its formulation. Although GA has taken larger population size and more generations, it often resulted in infeasible solutions. Only after many trials, it could yield feasible solutions, but these were still far below when compared to ACO results. The average hydropower productions obtained from RSOM-II in each month are shown in Figure 10. It can be seen that except in the months of July, August and October, in all the remaining time periods, ACO performance is significantly better than that of GA. The average storages of the reservoir at the beginning of each month are shown in Figure 11. In this case the average computational time requirements for ACO and GA are 32 min 28 sec and 80 min 54 sec respectively.

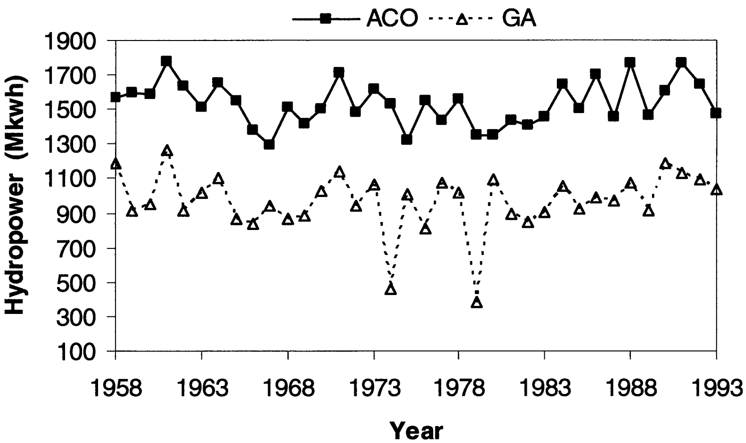

Figure 9. Annual hydropower production obtained using ACO and GA for RSOM-II.

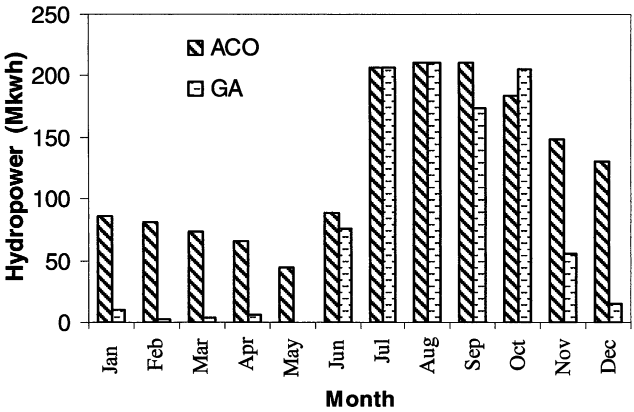

Figure 10. Monthly average hydropower production obtained using ACO and GA for RSOM-II.

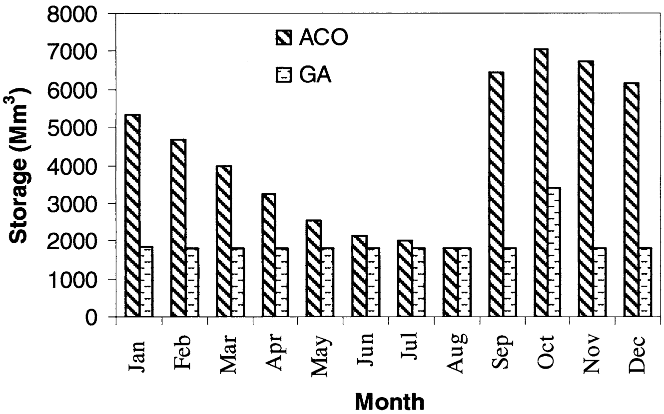

Figure 11. Monthly average initial storages obtained using ACO and GA for RSOM-II.

The results obtained clearly indicate that ACO performance is the better. Figure 12 shows the comparison of the results for both the short-time horizon operation (1 year at a time) and long time-horizon operation (36 years at a time) models. It can be seen that ACO model is performing better for long time horizon operation than for short-time horizon i.e., year wise operation models (both GA and ACO). Also it can be noticed that for long time period operation, ACO model is taking care of the carry over storage to meet the uncertainty of adequate inflow in the subsequent year and thus yielding better results. But in year-wise operation, this flexibility is restricted by the end of period storage constraint. So to derive reservoir operating policies for planning purpose and to analyze the policies with long term goals, the long time horizon operation will be more useful. Thus ACO performance will be very much useful in deriving such policies.

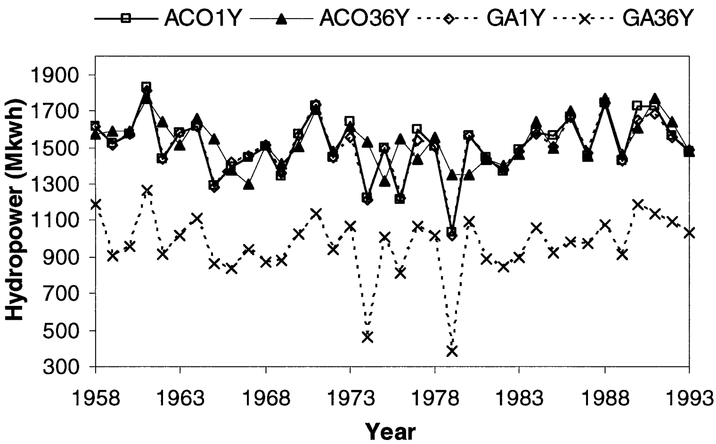

Figure 12. Comparison of annual hydropower production obtained for RSOM-I and RSOM-II using ACO and GA. (ACO1Y and GA1Y are results of RSOM-I, i.e., short-time horizon reservoir operation model. ACO36Y and GA36Y are results of RSOM-II, i.e., long-time horizon reservoir operation model)

From these results, the main advantages of ACO noticed are, even if the number of decision variables and constraints increase in the problem domain, the quality of optimal solution is not affected. Also to reach a near optimal solution, the number of ants and the number of iterations required are quite small and also the number of function evaluations required is less. One of the causes for the better performance of ACO, may be the randomized initialization of starting position of ants over the entire time horizon, thus enabling the algorithm to better explore the available solution search space. Also ACO uses heuristic information and probabilistic transition rules, to move from one storage class in the current time period to the next storage class in the next time period. This enables ACO to yield a better solution at each time step, consequently helping to find the optimal solution over the entire time horizon in a few iterations.

The results clearly show that in deriving reservoir operation policies for a long time-horizon, the over year storage requirement constraint can be relaxed, consequently which helps in better operation. Thus, in this regard the performance of a metaheuristics based algorithm like ACO is significant. This is a substantial improvement over other nonlinear optimization techniques, using which it is not possible to obtain optimal solution over such a long time horizon.

7. Conclusion

In this study a metaheuristic technique called Ant Colony Optimization (ACO) is applied for a reservoir operation problem and the usefulness of the technique is explored. The ACO algorithm for multi-purpose reservoir operation is presented using a single reservoir, deterministic, finite-horizon problem and applied to an existing reservoir system namely, Hirakud reservoir in India. In this model, power production is maximized after satisfying the flood control rules, meeting the irrigation demands and within the limitations of other physical constraints. After carrying a thorough sensitivity analysis for deciding values of concerned parameters, the ACO technique is applied to a short-time horizon reservoir operation model. To evaluate the performance of ACO model, the results are compared with those obtained by GA model. The results of both models showed closer matching. To further explore the potential of ACO, it is tested for a long-time horizon operation model. It is found that the reservoir operating policies developed by ACO give much better performance, yielding higher annual hydropower than by GA model. By optimizing the model for such a long time period, ACO facilitates to evolve better operation policies by considering the over year storage requirements. Thus it can be concluded that the ACO technique is quite promising and very much useful to derive efficient operating policies for a multi purpose-reservoir system.

References

Abbaspour, K. C., Schulin, R., and Genuchten, M. T. V., 2001, 'Estimating unsaturated soil hydraulic parameters using ant colony optimization', *Adv. Water Resour.* **24**, 827–841.

Chang, F. H. and Chen, L., 1998, 'Real-coded genetic algorithm for rule-based flood control reservoir management', *Water Resour. Manage.* **12**, 185–198.

Deb, K., 2000, 'An efficient constraint handling method for genetic algorithms', *Comput. Methods Appl. Mech. Engrg.* **186**, 311–338.

Dorigo, M., 1992, Optimization, learning and natural algorithms (*in italian*). Ph.D. thesis, DEI, Politecnico di Milano, Italy.

Dorigo, M. and Di Caro, G., 1999, 'The ant colony optimization metaheuristic', in D. Corne, M. Dorigo, and F. Glover (eds.), *New ideas in optimization*, McGraw-Hill, London, 11–32.

Dorigo, M., Di Caro, G., and Gambardella, L., 1999, 'Ant algorithms for discrete optimization', *Artif. Life.* **5**, 137–172.

Dorigo, M. and Gambardella, L. M., 1997, 'Ant colony system: a cooperative learning approach to the traveling salesman problem', *IEEE Trans. Evol. Comput.* **1**, 53–66.

Dorigo, M., Maniezzo, V., and Colorni, A., 1996, 'The ant system: optimization by a colony of cooperating ants', *IEEE Trans. Sys. Man Cybern.* **26**, 29–42.

Dorigo, M. and Stützle, T., 2004, *Ant Colony Optimization*. Cambridge, MA: MIT Press.

Goldberg, D., 1989, 'Genetic Algorithms in Search, Optimization and Machine Learning', Addison-Wesley Longman Publishing Co., Inc., Boston, MA.

Holland, J. H., 1975, *Adaptation in Natural and Artificial Systems*. University of Michigan Press: Ann Arbor, MI.

Jalali, M. R., Afshar, A., and Mariño, M. A., 2003, 'Reservoir Operation by Ant Colony Optimization Algorithms', online paper, http://www.optimizationonline.org/DB_FILE/2003/07/696.pdf, (accessed on 10/12/2004).

Maier, H. R., Simpson, A. R., Zecchin, A. C., Foong, W. K., Phang, K. Y., Seah, H. Y., and Tan, C. L., 2003, 'Ant colony optimization for design of water distribution systems', *J. Water Resour. Ping. and Manage. ASCE.* **129**, 200–209.

Michalewicz, Z., 1996, 'Genetic Algorithm +Data Structures = Evolution Programs'. 3rd ed. Springer-Verlag: New York.

Nagesh Kumar, D., Srinivasa Raju, K., and Ashok, B., 2005, 'Optimal reservoir operation for irrigation of multiple crops using genetic algorithms'. *J. Irri. and Drain. Engg., ASCE.*, in print.

Oliveira, R. and Loucks, D. P., 1997, 'Operating rules for multireservoir systems', *Water Resour. Res.* **33**, 839–852.

Proceedings of the 3rd meeting of the rule curve revision committee., 1988, Govt. of Orissa, India.

Sharif, M. and Wardlaw, R., 2000, 'Multireservoir system optimization using genetic algorithms: case study', *J. Comp. in Civ. Engrg. ASCE*. **14**, 255–263.

Stutzle, T. and Hoos, H. H., 2000, 'MAX-MIN ant system', *Future Generation Comput. Syst*. **16**, 889–914.

Wardlaw, R. and Sharif, M., 1999, 'Evaluation of genetic algorithms for optimal reservoir system operation', *J. Water Resour. Plng. and Manage. ASCE*. **125**, 25–33.

Wurbs, R. A., 1993. 'Reservoir-system simulation and optimization models', *J. Water Resour. Plng. and Manage. ASCE*. **119**, 455–472.

Yeh, W. W.-G., 1985, 'Reservoir management and operations models: a state-of- the-art review', *Water Resour. Res*. **21**, 1797–1818.

Zecchin, A. C., Simpson, A. R., Maier, H. R., and Nixon, J. B., 2005. 'Parametric study for ant algorithm applied to water distribution system optimization', *IEEE Trans. Evol. Comput*. **9**, 175–191.