Application of the dynamic ant colony algorithm on the optimal operation of cascade reservoirs

X X Tong¹,², W S Xu², Y F Wang², Y W Zhang³ and P C Zhang¹

¹ Changjiang River Scientific Research Institute of Changjiang Water Resources Commission, Wuhan, China

E-mail: ckytxx@163.com

Abstract. Due to the lack of dynamic adjustments between global searches and local optimization, it is difficult to maintain high diversity and overcome local optimum problems for Ant Colony Algorithms (ACA). Therefore, this paper proposes an improved ACA, Dynamic Ant Colony Algorithm (DACA). DACA applies dynamic adjustments on heuristic factor changes to balance global searches and local optimization in ACA, which decreases cosines. At the same time, by utilizing the randomness and ergodicity of the chaotic search, DACA implements the chaos disturbance on the path found in each ACA iteration to improve the algorithm's ability to jump out of the local optimum and avoid premature convergence. We conducted a case study with DACA for optimal joint operation of the Dadu River cascade reservoirs. The simulation results were compared with the results of the gradual optimization method and the standard ACA, which demonstrated the advantages of DACA in speed and precision.

1. Introduction

The reservoir optimal operation is a complex combinatorial optimization problem that is non-convex, nonlinear, and has high dimensions. Traditional algorithms, such as dynamic programming, successive approximation of dynamic programming, step-by-step optimization algorithm have obvious defects such as "dimension disaster", slow convergence speed and initial solution dependency, which have modest effect on joint dispatching of cascade reservoirs. Continued research development in biological intelligence, artificial intelligence, genetic algorithms, particle swarm optimization, and the differential evolution algorithm of intelligent optimization algorithms have been widely used in optimizing cascade reservoir operations.

Chen et al [1] proposed hypercube floating-point encoding and an adaptive adjustment control strategy of parameters to improve the genetic algorithm, and to enhance the local optimization ability
of the genetic algorithm. Zheng et al [2] proposed a method of chaos optimization to generate an initial solution; this approach controls parameters’ adaptive adjustment according to individual fitness and dispersion degree population, which overcomes premature GA and improves convergence performance. Ji et al. [3] introduced the catfish effect mechanism to the standard particle swarm algorithm; this particle swarm optimization algorithm (CE-PSO) improves the algorithm's ability to jump out of the local optimal solution. Qin et al. [4] brought the external file update strategy to the differential evolution algorithm and to modify operator, the improved algorithm can improve the uniformity of the distribution of population. Zheng et al. [5] embedded the hybrid leapfrog algorithm in the differential evolution algorithm, which can improve the global optimization ability of the algorithm through the interaction and evolution of the upper and lower double algorithm. Based on the optimal operation schedule made by the deterministic hydropower station optimal operation, Li et al. [6] used the idea of aggregation-decomposition and applied genetic programming to simulate the total output and the allocation of total output in cascade hydropower stations. This improved the cascade hydropower stations’ total output reliability. Although these algorithms have improved the optimal capacity of the original algorithm to some extent, they all failed to create dynamic balance between the global and local search.

The Ant Colony Algorithm (ACA) [7, 8] is an optimization method based on the swarm intelligence algorithm. This algorithm uses the walking path of artificial ants to represent the feasible solution of a specific problem, which makes use of the ant colony’s self-organization. It switches path information through pheromones to form collectively autocatalytic behavior and ultimately finds the optimal path. The algorithm has few control parameters and high robustness. However, this algorithm suffers from an imbalance between global search and local optimization, and the local optimal problems caused by quick reduce of population diversity in the process of evolution.

Therefore, this paper advances a new algorithm named Dynamic Ant Colony Algorithm (DACA) with improved parameter control and optimization strategies over ACA. DACA uses a cosine decreasing strategy and dynamically adjusts the heuristic factor to balance the ant global search and local optimization. At the same time, disturbing path found by each iteration of ant colony algorithm and using the randomness and ergodicity of chaotic search to improve the ability of local optimization algorithm and avoid local optimum.

2. Mathematical model
As a combined operation, cascade hydropower stations exhibit power, compensation, capacity compensation, and hydrological compensation. This paper establishes the optimal the long-term operation model of a cascade hydropower station to maximize energy output.

2.1. The objective function

\[ E = \text{Max} \sum_{t=1}^{T} \sum_{i=1}^{N} N_{it} \cdot \Delta t \]  

\[ N_{it} = A_i \cdot Q_{it} \cdot H_{i,t} \]  

where \( t \) is short for time; \( E \) is the total power’s generating capacity of the scheduling cascade; \( N_{it} \) is the output in period \( t \) of power station \( i \); \( A_i \) is the comprehensive efficiency coefficient of the power station in period \( t \); \( Q_{it} \) is the average output current of power station \( i \) in period \( t \); \( H_{i,t} \) is the
average power output head of power station \( i \) in period \( t \); \( T \) is the hours of the scheduling period; \( N \) is the number of cascade hydropower stations; \( \Delta t \) is the length of a single time period.

2.2. The constraint
- Water balance constraints
  \( V_{i,t} = V_{i,t-1} + (I_{i,t} - Q_{i,t} - q_{i,t}) \cdot \Delta t \)

- Water level constraints
  \( ZL_{i,t} \leq Z_{i,t} \leq ZU_{i,t} \)

- Runoff constraints
  \( QL_{i,t} \leq Q_{i,t} \leq QU_{i,t} \)

- Output constraints
  \( NL_{i,t} \leq N_{i,t} \leq NU_{i,t} \)

where \( t \) is time; \( ZL_{i,t} , \ ZU_{i,t} \) are the lowest and highest water level of power station \( i \) in period \( t \), respectively; \( QL_{i,t} , \ QU_{i,t} \) are the lower and upper limit of power runoff of power station \( i \) in period \( t \), respectively; \( NL_{i,t} , \ NU_{i,t} \) are the lower and upper limit of power output of power station \( i \) in period \( t \), respectively; \( I_{i,t} \) is the average inflow of reservoir \( i \) in the beginning and the end of period \( t \); \( q_{i,t} \) is the surplus water flow.

3. Ant colony algorithm

3.1. The basic principle
ACA is a heuristic random search method based on group collaboration. The artificial ants’ path attempts to solve the problem; the artificial ants choose the walk path according to the pheromones left behind by other hands. In doing so, they leave behind pheromones for subsequent ants, forming a positive feedback mechanism (Dorigo M et al. 1996). As the algorithm advances, pheromones in the optimal path increase while pheromones in the poor path decrease due to volatilization. Eventually, the whole ant colony would focus on the optimal solution under the positive feedback mechanism.

3.2. Ant colony algorithm in the application of the optimal reservoir operation
- Assume that reservoir water level at the end of each time period as decision variables, discrete the feasible domains of water level from normal storage level to dead storage level, that is to say, to solve the reservoir optimization scheduling model is to seek an optimal decision \( [x_1, x_2, \cdots, x_T] \) which can get maximum generating volume of cascade reservoirs. Therefore, the initial solution \( X = [X_1, X_2, \cdots, X_{NP}] \) is generated randomly within the scope of the
feasible solution, in which NP is the ant colony size. \( X = [X_1, X_2, \cdots, X_{NP}] \) is the route for ant \( k \); it represents the solution of the problem. \( D \) is the space dimension of the solutions.

- Generally, in the marching process of ant \( k \), the path selection is based on the principle of high pheromone concentrations in accordance with high transition probability. The probability calculation formula is as follows:

\[
p_{m,n}^k(t) = \begin{cases} 
\frac{[\tau_{m,n}(t)]^\alpha \cdot [\eta_{m,n}(t)]^\beta}{\sum_{j \in \text{allowed}_k} [\tau_{m,n}(t)]^\alpha \cdot [\eta_{m,n}(t)]^\beta}, & \text{if } j \in \text{allowed}_k \\
0, & \text{else}
\end{cases}
\]

(7)

where \( p_{m,n}^k(t) \) is the state transition probability of ant \( k \) from water level \( m \) to \( n \); \( \tau_{m,n}(t) \) is the residual amount of pheromones on path \((m,n)\) in stage \( t \). Each path during initialization pheromone amount are equal, is set as a constant; \( \eta_{m,n}(t) \) is the expected value of the transition amount from \( m \) to \( n \) in stage \( t \), referred to as prior knowledge; it is identified as the reciprocal of generating capacity \( d_{m,n} \) on the water level from \( m \) to \( n \), namely \( \eta_{m,n}(t) = 1/d_{m,n} \); \( \text{allowed}_k \) is the water level that ant \( k \) can choose in stage \( t \); \( \alpha \) is the inspired information factor, which reflects the relative importance of residual pheromones as the ant colony moves; \( \beta \) is the expected heuristic factor, which reflects the relative importance of expectation.

- To avoid expectation failure caused by excess residual pheromone, the pheromone left by each ant volatilizes and gradually decreases. The pheromone updating method is as follows:

\[
\tau_{m,n}(t + 1) = (1 - \rho) \cdot \tau_{m,n}(t) + \Delta \tau_{m,n}(t)
\]

(8)

\[
\Delta \tau_{m,n}(t) = \sum_{k=1}^{NP} \Delta \tau_{m,n}^k(t)
\]

(9)

where \( \rho \) is the volatilization factor; \( \Delta \tau_{m,n}(t) \) is the pheromone increment in this cycle on the path \((m,n)\) at initial moment \( \Delta \tau_{m,n}(t) = 0 \); \( \Delta \tau_{m,n}(t) \) is the amount of pheromone left by ant \( k \) on path \((m,n)\) in this cycle. The updating method of Ant Cycle Model in this article is:

\[
\Delta \tau_{m,n}(t) = \begin{cases} 
Q/L_k, & \text{ant } k \text{ passes through path } (m,n) \text{ in this circle} \\
0, & \text{other}
\end{cases}
\]

(10)
where $Q$ is the total amount of released pheromones by the ant on the path of one cycle or one process; $L_k$ is the path length of ant $k$ in this cycle.

4. Improved strategy and implementation steps of the ant colony algorithm

4.1. Dynamic adjustment strategy of heuristic factor

In general, the algorithm always has strong developmental ability in the early stage, and can search on the global scope; it increases its exploration abilities around the best solutions and improves local optimization abilities. The balance between global exploration and local optimization is an important aspect that affects the algorithm’s performance.

Formula 7 shows the ants’ routing probability. The heuristic factor $\alpha$ reflects the relative importance degree of pheromones accumulated by ants in guiding ants to choose the best path. The greater $\alpha$ is, the greater the influence of accumulated pheromones in the past is, and the greater the possibility that the ant chooses the past path. The weaker randomness would be of the search algorithm. The smaller $\alpha$ is, the lower the influence that the past path would have for the ant, and the more inclined it would be to perform a local search. Due to the same initial pheromone value in various paths, a much larger $\alpha$ would not significantly decrease the randomicity of the algorithm search. Accordingly, the value of $\alpha$ has dynamic adjustments in this paper; it uses a larger value early in the process to make the algorithm explore on a global scale, and it uses a smaller value later in the process to improve the local optimization ability. This study used a cosine decreasing strategy, to keep a smaller value early and a larger value later for $\alpha$, as follows:

$$\alpha = \alpha_{\min} + (\alpha_{\max} - \alpha_{\min}) \cdot \frac{\cos(\frac{\text{iter}}{\text{iter}_{\max}}) + 1}{2}$$

(11)

where $\alpha_{\max}$, $\alpha_{\min}$ are the upper and lower limits of the pheromone heuristic factor, respectively; $\text{iter}$, $\text{iter}_{\max}$ are the iterations times and the maximum iteration times, respectively.

Figure 1 shows the comparison of the cosine decreasing with linear decrease of $\alpha$.

![Figure 1](image.png)
4.2. Local chaos disturbance strategy

On the basis of the dynamic adjustment on heuristic factor, to avoid the local optimal solution of rapid aggregation in ACA, this paper utilized the ergodicity and randomness of a chaotic search and conducted local chaos optimization of ant colony in poor individual in the form of chaos disturbance, which improves the ability of ACA to jump out of local optimal solution and continue to find the optimal solution.

This study uses the one-dimensional logistic model to design chaotic mapping. The iterative equation is:

\[ R_{i}^{g+1} = c \cdot R_{i}^{g} \cdot (1 - R_{i}^{g}), \quad R_{i} \in [0, 1] \]  

where \( R_{i}^{g} \) is the chaotic variables, which represents a certain number generated by the chaotic equation between \([0, 1]\) and \( R_{i}^{1} \in [0, 1] \). \( g \) is the number of iterations; \( c \) is the control parameters and \( 0 \leq c \leq 4 \). When \( c > 3.57 \), \( R_{i}^{j} \in [0, 1] \) and \( R_{i}^{j} \notin \{0.25, 0.5, 0.75\} \), sequence will never repeated "wandering" in \((0, 1)\), not to any point of convergence, and sensitively depends on the initial point.

Assuming that the new ant colony \( X^{g} \) is obtained after iteration \( i \), disturb the ant colony according to formula (13) and (14), colony \( Y^{g} \) is got after disturbance. Where \( R^{g+1} \) is the matrix of \( D \) chaotic sequences and \( \gamma \) is the step disturbance.

\[
X^{g} = \begin{bmatrix} x_{1,1}^{g}, x_{1,2}^{g}, \ldots, x_{1,D}^{g} \\ x_{2,1}^{g}, x_{2,2}^{g}, \ldots, x_{2,D}^{g} \\ \vdots \\ x_{N_P,1}^{g}, x_{N_P,2}^{g}, \ldots, x_{N_P,D}^{g} \end{bmatrix} \quad R^{g+1} = \begin{bmatrix} R_{1,1}^{g+1}, R_{1,2}^{g+1}, \ldots, R_{1,D}^{g+1} \\ R_{2,1}^{g+1}, R_{2,2}^{g+1}, \ldots, R_{2,D}^{g+1} \\ \vdots \\ R_{N_P,1}^{g+1}, R_{N_P,2}^{g+1}, \ldots, R_{N_P,D}^{g+1} \end{bmatrix}
\]

\[ Y^{g} = X^{g} + \gamma \cdot R^{g+1} \]  

After obtaining the disturbance group \( Y^{g} \), based on the greedy selection strategy, compare the adaptation value of the corresponding individual \( X_{i}^{g} \) and \( Y_{i}^{g} \) of the original ant colony \( X^{g} \) and the disturbed ant colony \( Y^{g} \); take the individual with the better adaption value for the next generation of
ants, as follows:

\[ x_t^{g+1} = \begin{cases} x_t^g, & \text{if } f(x_t^g) \geq f(y_t^g) \\ y_t^g, & \text{if } f(x_t^g) < f(y_t^g) \end{cases} \]  \hspace{1cm} (15)

4.3. Steps for improved ant colony algorithm for solving joint optimization dispatching of cascade reservoirs

![Flowchart of steps.](image)

**Figure 2.** The flowchart of steps.

5. Case study

5.1. Introduction of cascade reservoirs and preferences of the algorithms
To validate the feasibility and effectiveness of the proposed DACA, and to apply the improved algorithm to solve the cascade reservoirs optimization scheduling problem, we conducted a case study on the cascade reservoirs of Dadu River. Figure 3 shows its cascade topology structure; table 1 shows the basic data of each plant. Reservoirs A and B have annual adjustment capabilities, while the other reservoirs only have daily adjustment capabilities; therefore, this multi-reservoir can be treated as a cascade reservoirs system with 2 reservoirs and 9 power stations.

![Figure 3. cascade topology.](image-url)

### Table 1. basic data of each power station.

| Item                | Unit | A    | A1   | A2   | A3   | B    | B1   | B2   | B3   | B4   |
|---------------------|------|------|------|------|------|------|------|------|------|------|
| Normal water level  | m    | 2930 | 2315 | 1692 | 955  | 850  | 660  | 528  | 474  | 432  |
| Dead water level    | m    | 2886 | 2307 | 1687 | 952  | 790  | 655  | 520  | 469  | 430  |
| Installed capacity  | MW   | 240  | 240  | 240  | 700  | 3600 | 660  | 730  | 600  | 480  |
| Warranted output    | MW   | 77.4 | 87.8 | 82.7 | 136.1| 926  | 162  | 197  | 138  | 151  |
| Efficiency coefficient | | 8.5  | 8.2  | 8.3  | 7.9  | 8    | 8.5  | 8.2  | 8.8  | 8.5  |

Take the measured runoff data as the input sequence of each month in a certain year. Use the three algorithm types, POA, ACA, and DACA, to optimally simulate the cascade reservoirs system. Refer to the parameter setting range of the Ant colony algorithm in the literature Intelligent Optimization Methods [9], and continuously adjust the parameter set to the test. Eventually, set the ACA algorithm parameter to the following: $NP = 200$, $\alpha = 5$, $\beta = 5$, $\rho = 0.3$, $Q = 5000$, $iter_{\text{max}} = 1000$. For DACA, $\alpha_{\text{max}} = 5$, $\alpha_{\text{min}} = 1$, $\gamma = 0.1$, and the other parameters are set to be the same as in the ACA.

For the POA, ACA, DACA three kinds of algorithms of the initial solution are randomly generated, the latter two algorithms in ant colony routing is according to certain probability to choose, also has a certain randomness. Therefore, compare that many times’ simulation results of the calculation of average results.

5.2. Calculation and discussion

Table 2 shows the calculation of the three algorithms, including the average total generating capacity and the average computing time.
The calculation found that ACA achieved \((364.5\times10^8\text{ kW} \cdot \text{h})\) total generating capacity, which was lower than that of POA at \((372.8\times10^8\text{ kW} \cdot \text{h})\). In the calculation process, ACA uses local optimization with an inadequate global search because of the increase of iterations times to quickly reduce ant colony diversity. Due to the strong guidance of the individual optimal path, the ant colony will be trapped in the local optimal value and cannot jump out. The cascade power generation solved by DACA \((374.9\times10^8\text{ kW} \cdot \text{h})\) is greater than those of POA \((372.8\times10^8\text{ kW} \cdot \text{h})\) and ACA \((364.5\times10^8\text{ kW} \cdot \text{h})\). This is because the DACA cosine decreasing strategy for introducing the heuristic factor can be as the iteration and the size of the dynamic adjustment heuristic factor, which makes the algorithm better balance the global search and local optimization. At the same time, chaotic disturbance strategies can strengthen the local mining algorithm ability, increase ant colony diversity, and improve the algorithm’s ability to jump out of local optimal value.

As shown in table 2, ACA had a 64s computing time, while DACA took 75s and POA took 310s. DACA’s computing time was slightly longer than that of ACA because DACA added the descending strategy and the ant colony stimulating factor of the cosine chaotic disturbance strategies. Considering the increased cascade power generation, DACA’s increased computing time is acceptable.

As for the algorithm structure, even though the POA algorithm only optimized the two stages each time, a single calculation amount will still exponentially increase with the increasing of number of reservoir (dimension), so the computing time must rise sharply. The calculation amount of ACA and DACA increased linearly with the efforts of the reservoir (dimension), so the influence of the power station number on the time-consuming was relatively small. To improve the quality of the ACA optimization result, DACA uses the cosine decreasing strategy of the heuristic factor chaotic disturbance strategies. As a result, increasing the reservoir quantity or the computing time has a slight effect on DACA performance.

The cascade reservoir water level was calculated by three algorithms, as shown in figure 4 and figure 5.

### Table 2. Different algorithms income average cascade total generating capacity and computing time.

| Algorithm | average total generating capacity \((10^8\text{ kW} \cdot \text{h})\) | average computing time (s) |
|-----------|-------------------------------------------------|---------------------------|
| POA       | 372.8                                           | 310                       |
| ACA       | 364.5                                           | 64                        |
| DACA      | 374.9                                           | 75                        |

The cascade reservoir water level was calculated by three algorithms, as shown in figure 4 and figure 5.
Observing the reservoir water level changes shown in figure 4, it is obvious that the POA and DACA water storage processes have significant differences. POA rapidly stores water early, resulting in more abandoned water. The early released water loses the high water head advantage. Meanwhile, DACA begins a water storage later, which takes full advantage of the large capacity, maintains high water for a longer period of time, and uses the high head to increase generating capacity. The design head of power station unit A is 560 meters, is the high water head, and flow rate of the change on the influence of the generating capacity is relatively more obvious, DACA’s calculation results is more suit for the power station feature than POA’s results.
The adjusted water level of reservoir B is relatively high, so it must increase the discharge to generate electricity in the early stages. At the same time, the unit design head is 148 m, the average flow rate for many years is 1230 m$^3$/s, so the effect of the water head change of the cascade total generating capacity is more apparent than the flow rate change. DACA’s generating capacity solution is higher than that of POA. DACA’s calculation result is more suitable for maintaining high water power generation.

6. Conclusions
In the proposed DACA, the cosine decreasing strategy of the heuristic factor is used to adjust the balance between global searches and local optimization of ACA. The randomness and ergodicity of the chaotic search are used to make chaos disturbance ant colony search results, and improve the ability to jump out of local optimal algorithm. The case study showed that the dynamic ant colony algorithm has stronger global searching ability and higher convergence stability than that of conventional algorithms.

Acknowledgements
The research work was supported by Open Foundation of Basic Scientific Research Operating Expenses of Central-Level Public Academies and Institutes (CKSF2016040/TB) and National Natural Science Foundation of China under Grant No.51209014.

References
[1] Chen L, Mei Y, Dong Y and Yang N 2008 Journal of Hydraulic Engineering 39(5) 550-56
[2] Zheng J, Yang K, Ni F and Liu G 2013 Journal of Hydraulic Engineering 44(2) 205-11
[3] Ji C, Liu F, Peng Y and Xiang T 2013 Journal of Hydroelectric Engineering 32(1) 70-6
[4] Qin H, Zhou J, Wang G and Zhang Y 2009 Journal of Hydraulic Engineering 40(5) 513-19
[5] Zheng H, Mei Y, Hu T, Huang C and Tang H 2013 Journal of Hydroelectric Engineering 32(1) 54-62
[6] Li L, Liu P, Zhang Z, Zhao J and Shen Y 2013 China Rural Water and Hydropower 2 134-7
[7] Dorigo M, Maniezzo V and Colomi A 1996 IEEE Transactions on Systems, Man, and Cybernetics – Part B 26(1) 29-41
[8] Dorigo M and Gambardella L M 1997 IEEE Transactions on Evolutionary Computation 1(1) 53-56
[9] Wang D, Wang J, Wang H, Zhang R and Guo Z 2007 Intelligent Optimization Methods (Beijing: Higher Education Press)