Journal of Mechanical Engineering Research and Developments
ISSN: 1024-1752
Website: http://www.jmerd.org
Vol. 38, No. 2, 2015, pp. 102-106

Overall Improved Genetic Algorithm Applied in Optimal Generation Dispatching of Multi-reservoir System

Y. F. Li[†*] & K. A. Tutu[‡]

[†]Sichuan University, Chengdu, 610065, China
[‡]Mendel University, Faculty of Regional and Territorial Studies, Brno, Czech Republics

ABSTRACT: This research first introduces the basic principles of genetic algorithms, and improves the algorithm by combining with self-adaptive optimization theory concerning on its shortage of low global searching ability and slow convergent rate, so that crossover operator and mutation operator can automatically change over the fitness, thus forming adaptive genetic algorithm (AGA) and maintaining the diversity of the population effectively; secondly, this paper introduces the concept of niche to form the ultimate niche adaptive genetic algorithm to improve the global search ability, to avoid falling into local optimal solution, and to accelerate the convergent rate. The results shows that the improved genetic algorithm can obtain better results, and provide an effective and viable new way in optimization of hydropower station.

KEYWORDS: Hydropower generation of multi-reservoir; Optimal generation dispatching; Genetic algorithm; Niche.

INTRODUCTION

After years of development and construction of hydropower, China has gradually formed multitudinous inter-basin, trans-regional large-scale hydropower stations. The research on optimal generation dispatching of multi-reservoir system mainly focuses on model building and algorithm. With the rapid expansion of the scale of hydropower, optimization models to build become more and more sophisticated, but the appropriate algorithms are either slow in computing speed, or easily falls into local optimization solution [1]. The genetic algorithm (GA) is a global random searching algorithm using the biological basic idea of natural selection and genetic mechanism for reference, with strong robustness [2, 3]. GA has great potentiality in planning and design of watershed large-scale system, and can be widely applied in optimal generation dispatching of multi-reservoir system [4-8].

Concerning on the poor convergence and premature problem of genetic algorithm in optimal generation dispatching of multi-reservoir system, researchers have made many improvements, for instance, trigonometric selection operator maintains the diversity of population and accelerates the convergent rate through the non-linear conversion of fitness function [9]; the introduction of simulated annealing in crossover and mutation as well as the idea of big mutation improve the convergence performance [10]. These improvements and other methods of improvement involve the whole process of genetic algorithm for optimal generation dispatching of multi-reservoir system. Given that the above mentioned research is not systematic, this paper targets at the characteristics of reservoir dispatching, sets the parameters of initial population coding and selection in consistency with optimal dispatching operation, and improves selection operator, crossover operator and mutation operator of the genetic algorithm to achieve performance enhancements.

PARAMETERS SETTING OF OPTIMAL GENERATION DISPATCHING BASED ON GENETIC ALGORITHM

In order to solve the existing problems of genetic algorithm, this part makes reasonable setting of parts of the algorithm.

Settlement of Coding Method

In optimizing real variables, this paper adopts real coding as possible and follows the principle of minimum optimization variables. The less optimization variables are, the smaller the searching space is, thus the algorithm is easier in convergence. When using genetic algorithm to solve optimization problems, if certain constraints are taken into account in the encoding process, the new generated individuals still satisfy these constraints even with their parents going through operations of crossover and mutation. Therefore, there is no need to make supernumerary and impractical calculation, thereby improving the computational efficiency of the algorithm.

In real coding method, the genetic value must be within the given interval limits. The genetic operators such as crossover and mutation used in the genetic algorithm also must ensure that the new individual gene value produced by calculation is within the limits of the interval. Furthermore, when using more than one byte to represent a genetic value, crossover must be carried out in the boundary byte of two genes, and cannot be processed in the separation of a byte of a particular gene. The real coding has the following advantages.

1) Real coding of genes avoids the risk that the possible solution with better adaptive value in the searching space cannot be represented due to poor coding accuracy.

2) Real coding saves the decoding process in binary encoding. It is enough to design the specific genetic operators in correspondence to real coding.

3) The genetic algorithm acting on the real-coded gene has the ability of gradient change using continuous variable function.

4) Real coding can eliminate "Hamming cliff". In adoption of simple binary encoding, a large Hamming distance makes it difficult to find the optimal point even the algorithm has searched around it.

Setting the Initial Population

Generally there are two methods of generating initial population, one is completely random generation method, the other one is to use some heuristic algorithms or experiences to choose better individuals as initial population. However, both methods have disadvantages. The former cannot ensure the feasibility of the individual randomly generated, resulting increased evolutional generation in complex optimization problems, which leading to poor convergence and low solving efficiency. The latter might lead to precocity and unable to find the global optimal solution due to the unrepresentative chosen chromosome.

This paper combines the advantages of these two methods, and chooses random individuals, empirical individuals and interfacial individuals according to certain proportion to form the complete initial population.

Random individuals: according to the known data and a variety of constraints, a group of initial individuals are randomly generated, as a major part of the initial population.

Empirical individuals: feasible empirical individuals are made by experienced dispatchers based on dispatching data through previous years and their own experience.

Interfacial individuals: when the population is filled by a non-global optimal individual or individuals close to this individual, genetic algorithm will fall into local optimum, namely the phenomenon of precocity. In order to avoid precocity and increase the diversity of individuals, an appropriate number of interfacial individuals are introduced through the feasible solution found in boundaries of feasible region, in order to ensure the diversity of the population.

Fitness Processing

The optimal regulation of reservoirs is the solution of maximum value of the optimized objective function. The individual fitness of genetic algorithm can be directly represented by the correspondent objective function plus penalty function. Through this processing, the optimization problem with constrains can be transferred to the optimization without constraints taking penalty function as objective function.

IMPROVEMENT OF GENETIC OPERATORS

In the genetic algorithm, optimization ability and the final result are mainly determined by genetic operation. Therefore, the improvement of genetic algorithm mainly done from the selection operator, crossover operator and mutation operator. This paper makes appropriate choices and improvements of selecting operation and mutation operation combining with the research of optimal generation dispatching of multi-reservoir system.

Selecting Operation

Selecting operation uses the ranking order selection method, and makes appropriate improvements. The basic principle is to arrange the group fitness in descending order, the previous $M \bullet \mathrm{p}_r$ (M is the group scale, p_r is the copy probability) individuals are directly copied into the next generation, then $M \bullet (1-\mathrm{p}_r)$ individuals from the overall group are chosen randomly in the way of roulette into the next generation. By doing so, under the assurance of constant population size, the excellent individuals are ensured to enter into the next generation, meanwhile the diversity of the population is ensured too.

Selection of Crossover Operator

In this paper, single-point crossover is adopted. The following are the detail operation. Set an intersection randomly in the individual gene string. In the implementation of the crossover, part of the structure of two individuals before or after that intersection will be exchanged, and generate two new individuals. For example, parent of floating point array:

$$\textit{Before the crossover}\begin{cases} parent:(23,10,17,\cdots,31,36,38,41) \\ child:(19,25,32,\cdots,27,20,13,12) \end{cases}$$

Take the last but three as an intersection randomly, and there generates two new individuals:

$$\textit{Before the crossover}\begin{cases} parent:(23,10,17,\cdots,31,20,13,12) \\ child:(19,25,32,\cdots,27,36,38,41) \end{cases}$$

Selection of Mutation Operator

This paper adopts non-uniform mutation for the following reasons. The uniform mutation takes random numbers uniformly distributed within a range to replace the original gene values, although this enables an individual to move freely within the searching space, but on the other hand it is not easy for local search in certain key areas. To improve this performance, this paper makes random perturbations to the original gene values, and takes the results as the new gene value after mutation. After mutation operation of each gene locus with the same probability, the entire solution vectors go through a slight change in the solution space. This mutation method is called non-uniform mutation.

Assume an individual $X = x_1 x_2 \cdots x_m \cdots x_1$, after mutation the individual is $X' = x_1' x_2' \cdots x_m' \cdots x_1'$, the mutation operation is:

$$x_m' = \begin{cases} x_m + (\mathrm{x}_m^{\max} - \mathrm{x}_m) * \mathrm{p}_m * \mathrm{C(g)} \\ x_m - (\mathrm{x}_m - \mathrm{x}_m^{\min}) * \mathrm{p}_m * \mathrm{C(g)} \end{cases} rd(0,1) = 1$$

In the formula, x_m and x_m' stand for the m[th] gene value of the individual before mutation and after mutation respectively; $\mathrm{x}_m^{\max}$ and $\mathrm{x}_m^{\min}$ are the upper limit and lower limit of gene values respectively; p_m is adaptive probability of mutation; $\mathrm{C(g)}$ is the parameter of evolutionary control; $rd(0,1) = 1$ is the equal probability binary random generating function. p_m and $\mathrm{C(g)}$ will be explained in the following part.

ADAPTIVE CROSSOVER OPERATOR AND MUTATION OPERATOR

Adaptive crossover probability p_c and mutation probability p_m

After studying the adaptive methods of genetic probability, Srinivas proposes the adaptive genetic algorithm (AGA). Its basic idea is to make the fitness of crossover and mutation probability change automatically. When the fitness of each individual in the population tends to converge or in case of regional local optimum, p_c and p_m increase. When the fitness of the population scatters, p_c and p_m decrease. Meanwhile, the individual whose fitness is higher than the average fitness of the population, has lower p_c and p_m so that this solution can be protected into the next generation. And the individual whose fitness is lower than the average fitness of the population, has higher p_c and p_m, which means this solution will be eliminated. Therefore, adaptive p_c and p_m can provide the optimal p_c and p_m of certain solution. This algorithm maintains the diversity of the population as well as ensures the global convergence of the genetic algorithm.

The variation curves of crossover and mutation probabilities in adaptive genetic algorithm are shown as (a) and (b) in Figure 1 respectively.

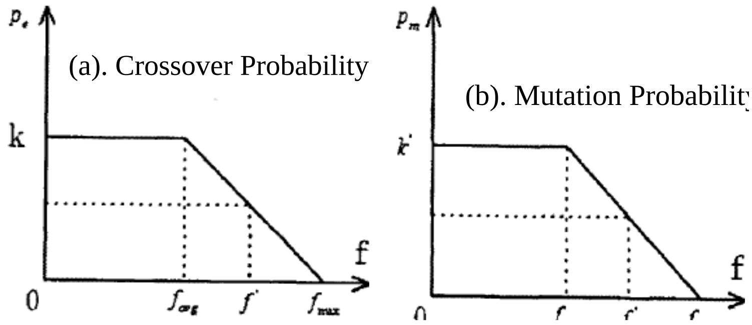

Figure 1. Variation curves of probability.

Through the comparison of individual fitness with the overall performance of the population, adaptive p_c and p_m determine their crossover and mutation probabilities, which are calculated as follows:

$$p_c = \begin{cases} \dfrac{k_1(\mathrm{f}_{\max} - \mathrm{f}^{'})}{(\mathrm{f}_{\max} - \mathrm{f}_{avg})} (f^{'} \geq f_{avg}) \\ k_2 \end{cases} \quad p_m = \begin{cases} \dfrac{k_3(\mathrm{f}_{\max} - \mathrm{f}^{'})}{(\mathrm{f}_{\max} - \mathrm{f}_{avg})} (f^{'} \geq f_{avg}) \\ k_4 \end{cases}$$

In the formula, $\mathrm{f}_{\max}$ is the maximum of the group fitness, f_{avg} is the average for the group fitness， $f^{'}$ is the larger fitness of the two individuals possible of crossover, and $k_1, k_2, k_3, k_4 \in [0,1]$.

If the fitness is below average fitness, it indicates that the individual is in bad performance, as for it, larger crossover and mutation probabilities will be applied; if the fitness is above average fitness, it indicates that the individual is in excellent performance, as for it, the corresponding crossover and mutation probabilities are chosen based on its fitness. As can be seen, when the fitness value is closer to its maximum, crossover and mutation probabilities are smaller; when the fitness value is equal to the maximum, the crossover and mutation probabilities are zero. This adjustment method is more suitable for groups in the late evolution, but unfavorable for the early evolution. This is because in the early evolution, the excellent individuals in the group almost are in a state that does not change, and are not necessarily the global optimum at this time, which increases the possibility of local optimal solution. This is why this paper chooses to integrate the empirical individuals, random individuals and interfacial individuals in generating initial population.

Parameter of Evolutionary Control $\mathrm{C(g)}$

The function of $\mathrm{C(g)}$ is to ensure the convergence of genetic algorithm, so that the mutation step decreases with the increase of evolutional generation, to avoid turning genetic algorithm into random search. Its formula is shown as below.

$$C(g) = 1 - r^{(1-g/G)C}$$

In the formula, g is the evolutionary generation of the population, r is the uniformly distributed random number within range of $[0,1]$, G is the maximum of evolutionary generation, C is the system parameter with general value of 2.

NICHE TECHNOLOGY TO IMPROVE GA

The basic genetic algorithm has the problems of premature convergence and slow convergence in later stage, which makes the final optimization results are often less than ideal results. Therefore, the key to complex optimization problems with multi-constrains using genetic algorithm is to find the global optimal solution quickly and prevent precocity and convergence phenomenon. Moreover, in order to ensure global convergence of genetic algorithm, the diversity of individuals in the population must be maintained to avoid the loss of individuals with high fitness; while in order to speed up the convergence, it is necessary to make the population shift to the optimal state as quick as possible, which will reduce the diversity of the population.

To solve these problems, this paper proposes an adaptive niche genetic algorithm for solving global searching performance and issues of convergence speed. Here is the basic idea. Firstly, calculate the Hamming distance between each two individuals. If the distance is less than the preset distance L, impose a greater penalty function to the individual with low fitness to reduce its fitness greatly. The individual with bad fitness becomes worse after this processing, and will be eliminated in the later evolutionary process in extremely large probability. This method is to protect the diversity of the population, to avoid a large number of duplicate solutions flooded the solution space. However, niche genetic algorithm has slow convergence, poor local optimization, and is prone to see evolutionary stasis. Therefore, by combining the adaptive crossover operator and mutation operator with niche, this paper forms adaptive genetic algorithm based on niche technology, improves the performance of the algorithm from all aspects, and improves the feasibility of the algorithm effectively.

CONCLUSION

This paper makes adaptive improvement of the selecting operator, crossover operator and mutation operator of the genetic algorithm, so that the improved operators can adjust the crossover and mutation probabilities automatically based on fitness function. Then this paper determines the initial population, parameters of coding and selection in line with the optimal scheduling algorithm. Finally, this paper applies the concept of niche into the genetic algorithm, which enhances the global searching ability of genetic algorithm.

REFERENCES

[1] X. Dong, D. Z. Xu, and Y. F. Jiang, "Research on Reservoirs Dispatch for Beneficial Purpose Based on Genetic Algorithm-A Case Study of Guanyinge Reservoir, Shenwo Reservoir and Tanghe Reservoir", *Water Conservancy Science and Technology and Economy*, vol. 17, no. 9, pp. 76-78, 2011.

[2] K. Li, X. Y. Ma, and S. H. Fu, "Scheduling Optimization Model and Algorithm Based on Improved Genetic Arithmetic for Hydropower Station", *Water Power*, vol. 36, no. 1, pp. 92-96, 2010.

[3] H. Lu, S. L. Chen, F. F. Yang, and Y. T. Li, "Optimizing Reservoir Operation Chart Based o n Time- of- use Electricity Pricing", *Water Resources and Power*, vol. 29, no. 12, pp. 46-49, 2011.

[4] J. Zheng, K. Yang, Y. H. Hao, R. Zhou, and G. S. Liu, "Application of Steady-state Genetic Algorithm to Optimal Operation of Reservoir", *Water Resources and Power*, vol. 29, no. 8, pp. 38-41, 2011.

[5] F. Wan, Q. Huang, W. L. Yuan, and L. Qiu, "Research on Co-Evolutionary Genetic Algorithm for Reservoir Optimization Water Supply Dispatching", *Journal of Xi'an University of Technology*, vol. 27, no. 2, pp. 139-144, 2011.

[6] J. Shu, B. Han, and L. Z. Zhang, "Self- scheduling of Cascaded Hydropower Stations Based on Pheromone Induction Genetic Algorithm", *Journal of Hydroelectric Engineering*, vol. 30, no. 2, pp. 32-37, 2011.

[7] P. A. Zhong, B. Xu, and J. H. Zhang, "Improvement of Genetic Algorithm for Its Application to Optimal Operation of Hydropower Station", *Journal of Hydroelectric Engineering*, vol. 30, no. 5, pp. 56-60, 2011.

[8] J. Zou and Y. Q. Zhang, "Rectangular Coding Genetic Algorithm for Cascade Reservoirs Operation", *Journal of Hydroelectric Engineering*, vol. 31, no. 1, pp. 27-31, 2012.

[9] K. Yang and J. Zheng, "Application of Genetic Algorithm with Trigonometric Selective Operators in Optimization of Cascade Reservoirs Operation", *Journal of Tianjin University (Science and Technology)*, vol. 45, no. 2, pp. 167-172, 2012.

[10] G. Li, J. Zou, and B. Zhang, "The Genetic Algorithm Simulated Annealing of Large Probability of Mutation and Its Application in Reservoir Optimization", *China Rural Water and Hydropower*, no. 3, pp. 148-151, 2010.