A Golden Section-based Double Population Genetic Algorithm Applied to Reactive Power Optimization

Zhongyong Wang¹, Yingjing Xu²*

¹ School of Automation, Guangdong University of Petrochemical Technology, Maoming City, Guangdong Province, 5252000, China
² Academic Affairs Office, Guangdong University of Petrochemical Technology, Maoming City, Guangdong Province, 5252000, China
*Corresponding author’s e-mail: xuyingjing@gdupt.edu.cn

Abstract. In this paper, a more effective method-the improved double population genetic algorithm is applied to reactive power optimization. The proposed algorithm is aimed at optimization strategy which makes the population exploit in new space successively. Premature convergence and weak local optimization are two key problems existing in the conventional genetic algorithm. Based on the golden section, the double population genetic algorithm (DPGA) is optimized and applied to the reactive power optimization in power system. Numerical simulation results demonstrate the validity of the proposed design algorithm.

1. Introduction
Power optimization is a high-dimensional, non-linear problem with the complexity of space-time coupling, continuous and discrete variables. With the development of power system, reactive power optimization problem is becoming more difficult than before [1][2][3]. For its practical application value, the reactive power research got much attention. In recent years, with the development of the intelligent theory, fuzzy and artificial neural network is applied to determine the control strategies of transformer taps and capacitors, but it is difficult to optimize quickly due to the increasing dimension of control variable [4][5][6]. Genetic algorithm is applied to solve the problem with adding the regulation times into objective function as penalty terms, and the penalty factors which have significant impact on the performance of algorithm are difficult to determine. In [7][8], considering the regulation times as restriction, the total energy loss of dispatching period is optimized.

Genetic algorithm is a subset of evolutionary algorithms that model biological processes to optimize highly complex cost functions [9][10][11][12][13]. Genetic algorithm is inspired from the mechanics of natural selection and natural genetics such as inheritance, mutation and recombination (also known as crossover). It is used in computing to find the optimal solutions to optimize and search problems. Genetic algorithm was developed by John Holland in 1975 over the course of 1960s and 1970s. Though the work of John Holland in development of genetic algorithm is most significant, several other scientists were also involved in developing similar theories and algorithms. The genetic algorithm differs substantially from the more traditional search and optimization methods.

The golden section is a ratio that exists in the growth patterns of many biological systems, such as spirals formed by shells or curves of ferns. The golden section was derived by the ancient Greeks, its ratio is an irrational number 0.618.
The optimized genetic algorithm is applied to reactive power optimization in this paper, the improved crossover operation is in possession of the ability of fast local adjustment, the improved mutation operation combines sensitivity analysis to generate new individuals.

In the end, the proposed approach is examined in one testing system, the numerical simulation results demonstrates that the proposed method can effectively decrease network loss.

2. GA Mechanism Analysis
Genetic algorithm can rapidly find a good solution; Genetic algorithm searches the solution space multiple points in parallel; Genetic algorithm is based on probabilistic transition rules and not deterministic ones. Genetic algorithms have a tendency to converge prematurely to a non-optimal point.

Operation of choice
Supposing the population space 
\[ S = \{ (X_1, X_2, \ldots, X_N) \} \quad X_i \in S, i = 1,2 \cdots N \]  
(1)
According to the rules of probability, there will be
\[ P(X_i) = \frac{f(X_i)}{\sum_{k=1}^{N} f(X_k)} \]  
(2)
After choosing arithmetic operators, the vector probability distribution is as follows:
\[
\begin{bmatrix}
X_1 & X_2 & \cdots & X_N \\
P\{X_1\} & P\{X_2\} & \cdots & P\{X_N\}
\end{bmatrix}
\]  
(3)
Where
\[ P\{X_i\} > 0; \sum_{i=1}^{N} P\{X_i\} = 1 \]  
(4)
The choice of arithmetic operators is merely within a fixed population and it is impossible to choose better individuals beyond this population. What’s more, it can be known from the selection process that it is easier for those individuals that have better adaptability to be chosen while it is easier for those individuals that have little adaptability to be eliminated. As a result, through the continuous selection among a fixed population, those individuals that have better adaptability will recur continuously and repeatedly and in this way, it is easier for the population to become premature.

Cross-operation:
Suppose D is the coding subsets of the coding space S,
\[ D = \{ X(x_1, \ldots, x_N) \in S \}; i = 1,2 \cdots N \]  
(5)
Also
\[ 1 \] is the length of strings of data in binary system; \( B(x) \) is the degree of maturity, \( X \) denoting the numbers of those individuals with the same values; \( \{ X(k), k \geq 0 \} \) means that the Markov chain correspondent to GA variation rate \( P_m = 0 \). Assume that \( X(0) = X_0 \)
then
\[ \forall \varepsilon \in D(x_{i1}, \ldots, x_{iB(x)}), X(0) \]  
(6)
If \( X(0) \) exists, then
\[ P\{ \varepsilon \in X(n) \mid X(0) = X_0 \} > 0 \quad n > 0 \]  
(7)
The formula (7) means that the crossed arithmetic operators have the ability to search for all individuals in the current population of minimal modes.
\[ \forall \varepsilon \not\in D(x_{i1}, \ldots, x_{iB(x)}), X(0) \]  
(8)
the result will be got as follow:
The formula (9) illuminates that the searching ability of the crossed arithmetic operators is only limited to the current minimal modes.

From the analysis above, the crossed arithmetic operators have premature domino effects during the searching process. Meanwhile, it is unavoidable that the crossed arithmetic operators will lessen the population diversity and thus the searching scope, causing its premature constringency.

Then variations of the model can increase the population modes but the variation rate should be controlled within a very small scope. Otherwise, lots of good modes will be destroyed and thus the arithmetic method will become a random search.

Therefore it can be seen that due to the limitation of the population dimensions and the mechanism of selection and crossed arithmetic operators, the traditional GA is very easily got the result in prematurity and partial optimization.

3. DPGA of Golden Section Optimized

3.1 Golden Section

Golden Section is a mathematical scale relations familiar to people. The so-called Golden Section rate is to get a point X on a unit length line segment and make the proportion of the longer length and the unit length equivalent to that of the shorter length and the longer one. That is

$$\frac{x}{l} = \frac{l-x}{x} \Rightarrow x^2 + xl - l^2 = 0 \Rightarrow x = 0.618l$$

0.618 is often called as the Golden Section ratio. The Golden Section ratio means the internal harmony and balance of things and full embodiment can be found in the biological circles. For such reasons we introduce the law of 0.618 in the improvement of genetic algorithm in order to raise the efficiency of operation.

DPGA is an improved arithmetic method based on the traditional genetic algorithm. It does genetic operations through the two parallel populations with the same dimensions but working separately on their own. One of the populations is chosen as the main population and the other as the secondary one. After a respective genetic operation of these two populations (hereditary, overlapping and variations), a new population will be generated. The individuals in the populations will be classified according to the value of its adaptability and the partial exchange in the individuals will be made between the two populations according to the Golden Section rate. Then the two populations will move to the next genetic operation respectively until the main population satisfies the softening conditions what the next generation needs.

3.2 Steps for the operation

By the continuous updating of the secondary population and partial assimilation to the main population and the joining of a fixed new mode to the main population in real time, the searching ability and scope of algorithm will be enhanced. Even if the main population is partially optimized and a disturbance will appear as a result of the impact created by the joining of the new individuals to the secondary population, the main population will step out of the partial optimization and the overall optimization will be achieved eventually. Because that it is double-population evolution, we can learn from its other useful improvements on algorithm and introduce a different algorithm into the two populations receptively in order to find out the optimized answer by algorithm as soon as possible.

The genetic operation process based on double population genetic algorithm is as follows:

**Step1.** Initialized population: Within the range of possible answers to the question, the two populations $P_1, P_2$ (P1 is the main population and P2 is the secondary population) will be initialized and one of them will be chosen as the main population and the answer space will be projected into the genetic space by a certain coding method and the coding length will be determined.

**Step2.** Genetic operation: $P_1, P_2$ will be under different (or same) genetic evolutional operation
and the new populations $P_{Ⅰ}^i$, $P_{Ⅱ}^i$ will be obtained.

**Step3.** The calculation of the degrees: Calculate the degrees of individual adaptability in $P_{Ⅰ}^i$, $P_{Ⅱ}^i$ and judge whether they satisfy convergence conditions. If they do (7) will be implemented.

**Step4.** Migration: From the 0.618N individuals of the values of adaptability in $P_{Ⅱ}^i$, previous 0.382N individuals will be chosen according to gambling round laws and they will be exchanged with 0.382N individuals afterwards.

**Step5.** Increase population modes: From a feasible range 0.618N individuals will be generated randomly. They and the 0.382N individuals exchanged from $P_{Ⅰ}^i$ will form a new population. from $P_{Ⅱ}^i$.

**Step6.** Convergence judgment: Repeat the above steps from (2) to (5) until the conditions of convergence of algorithm are satisfied.

**Step7.** Output results;

**Step8.** The end .

Then the basic flow chart of the algorithm analysis is presented:

![Fig 1. Block Diagram For Double Population](image)

### 4. Applied to Reactive Power Operation Model

#### 4.1. Reactive Power Operation Model

The practical load of power system is continuous variable, but the continuous load is not fit for reactive power optimization. Usually, the continuous load curve is simplified into time-interval trapezoid distribute curve, and it is consider that the load is constant in each time-interval.
Control variables such as generator voltages, transformer taps and capacitors must be optimized in order to minimize energy loss and make sure bus voltages reasonable.

The objective function is:

\[
\min P_L = \sum_{k \in N_e} P_{KL} = \sum_{i \in N_g, j \in N_t} G_{ij} (V_i^2 + V_j^2 - 2V_iV_j \cos \theta_{ij})
\]  
(11)

Power flow equation constraints are:

\[
P_{Gi} = P_{Di} + V_i \sum_j V_j (G_{ij} \cos \theta_{ij} + B_{ij} \sin \theta_{ij}), \quad i \in \{N_{pv}, N_{av}\}
\]  
(12)

\[
Q_{Gi} = Q_{Di} + V_i \sum_j V_j (G_{ij} \sin \theta_{ij} - B_{ij} \cos \theta_{ij}), \quad i \in \{N_{pv}, N_{av}\}
\]  
(13)

Also

\[
V_i^{\min} \leq V_i \leq V_i^{\max}, \quad Q_{Gi}^{\min} \leq Q_{Gi} \leq Q_{Gi}^{\max}
\]  
(14)

Where \( P_{Gi}, Q_{Gi} \) are the active and reactive power of the generator in bus \( i \); \( P_{Di}, Q_{Di} \) are the active and reactive power of the load in bus \( i \); \( V_i, V_j \) are the voltages of bus \( i \) and \( j \) respectively; \( G_{ij}, B_{ij} \) are the element of admittance matrix; \( \theta_{ij} \) is the phase difference between bus \( i \) and \( j \); \( n \) is the number of buses; \( P \) is the energy loss.

When the generation algorithm is applied to solve the above optimization problem, the power flow equation constrains are satisfied automatically. From Above the objective function is got as follows:

\[
\min F = P_0 + \lambda_G \sum (V_i - V_i^{\text{lim}})^2 + \lambda_Q \sum (Q_{Gi} - Q_{Gi}^{\text{lim}})^2
\]  
(15)

Where \( \lambda_G, \lambda_Q \) are penalty factors.

### 4.2 Simulation Results and Analysis

The effectiveness of the proposed control algorithm can be evaluated by the difference results from the Genetic Algorithm.

The simulation results are obtained under some assumptions as follow,

| Type         | Location | Quantity   | Control Range |
|--------------|----------|------------|---------------|
| Regulating transformer | 2        | (5,6) (4,3) | 0.90-1.10     |
| Capacitor banks | 2        | (4,6)      | 1-10          |
| Generator    | 2        | (1,2)      | 0.95-1.10     |

The parameters of the algorithm are chosen as
\( \lambda_G = 5, \lambda_{\alpha} = 10, \lambda_{\beta} = 15, M_1 = 100, M_2 = 70, N = 100, N_p = 5, P_c = 0.45, P_m = 0.02, \)
\[ T_{\text{step}} = 0.025, U_{\text{step}} = 0.01, Q_{\text{step}} = 1 \]

For the objective function, optimization is carried by the traditional genetic algorithm and the algorithm introduced in the paper. The comparison results show as below:

| Algorithm            | Golden Section-Based DPGA | Traditional GA |
|----------------------|---------------------------|----------------|
| Iteration times      | 96                        | 50             |
| Computing time       | 0.98                      | 0.71           |
| Active power loss    | 0.0936                    | 0.0779         |
| Reactive power loss  | -0.2862                   | -0.2810        |
| Bus maximum voltage  | 1.0319                    | 1.0546         |
| Bus minimum voltage  | 0.9618                    | 0.9680         |

From the results, it is clear that algorithm considered in this article is competitive in optimizing the reactive power optimization. It was found to be relatively better than the traditional genetic algorithm.

5. Conclusion
Reactive power optimization is a high-dimensional, non-linear, non-convex problem with the complexity of space-time coupling, continuous and discrete variables. A novel method, double population genetic algorithm based on golden section, is presented in the reactive power optimization for a power system. The simulation results verify the feasibility and validity of the proposed optimization method.

Acknowledgement
Project funding:
   (1) General Project of NSFC (October 2019 to September 2022), Research on Key Technologies of Data Driven Petrochemical Process Fault Safety, No. :61973094;
   (2) Nonlinear Control of flywheel system based on mathematical model, Guangdong University of Petrochemical Technology “Talent Introduction Project”, No.: 2017re14.

References
[1] ZHANG Li-zi, SHU Jun, LIN Xian-shu, et al. (2000) Reactive Power Planning Based on Genetic Algorithm. Proceedings of the CSEE, 20(6):5-8.
[2] Liang Caihao, Duan Xianzhong, Zhong Zhiyong, et al. (2006) Parallel reactive power optimization based on differential evolution and PC-cluster. Automation of Electric Power Systems, 30(1):29-34.
[3] Zhang Fengtian, Song Jiahua, Li Jian, et al. (2007) A hybrid differential evolution method for optimal reactive power optimization. Power System Technology, 31(9):33-37.
[4] LI Yu-long, ZONG Wei, Lü Xian-yan, et al. (2008) An Immune Genetic Algorithm Based on Regulating New Definition of Antibody Density for Power System Reactive Power Optimization. Electric Power, 23(2):115-119.
[5] JIANG Hui-ying. (2006) Application of Immune Genetic Algorithm in Reactive Power Optimization. Electro-technical Application, 25(4):67-69.
[6] Wu Q H, Cao Y J, Wen J Y. (1998) Optimal Reactive Power Dispatch Using an Adaptive Genetic Algorithm. Int J of Electrical Power & Energy Systems, 20(8):563–569.

[7] SHE Wei, LIU Jian, ZHOU Yan. (2009) Power systems reactive power regulation considering schedule optimization. Automation of Electric Power Systems, 33(4):31-35.

[8] ZHANG Deng-ke, HUANG Zhi-xiong. (2008) Reactive power optimization based on stochastic particle swarm optimization. Power System Protection and Control, 36(12):35-38.

[9] Yang Zhimin, Wang Xu, Zhuang Xianyi. (2000) Survey of Genetic Algorithm’s Application in Field of Automatic Control, Information and Control, 29(4): 329-339.

[10] Zhang Ling, Zhang Bo. (2000) Research on the Mechanism of Genetic Algorithms, Journal of Software, 11(7):945-952.

[11] Ding Chengmin, Zhang Chuansheng, Liu Hui. (1997) Discussions on some issues in Genetic Algorithms, Information and Control, 26(1): 40-47.

[12] Jiang Liying, etc. (2001) Self-optimizing parameters of PD Controller Based on Elitist Preserved Genetic Algorithm, Journal of Devices and Instruments, 22(3) Additional: 45-46.

[13] Xiong Weiqing, Wei Ping, Zhao Jiellyu. (2001) Research on the premature convergence of Genetic Algorithms, Computer Application Research, 9:12-14.