

# Improved NSGA-II Algorithm for Optimization of Constrained Functions

Yun Zhang\* and Bin Jiao

Shanghai Dianji University Shanghai, China

\*Corresponding author

**Abstract**—In order to solve the constrained multi-objective optimization problem, an improved NSGA-II algorithm is proposed. On the basis of NSGA, the cross operation of the feasible and unfeasible solution is implemented in order to give full play to the role of the infeasible solution in the optimization process. In addition, the external preservation set is updated on the basis of the obtained dominant individual to preserve the optimal solution of the problem. The improved algorithm is applied to typical test functions and compared with NSGA-II. The experimental results show that the algorithm is superior.

**Keywords**—multi-objective optimization; improved non-dominated sorting genetic algorithm; infeasible solutions; external save set

## I. INTRODUCTION

In production management, job scheduling, metallurgical process, and other engineering fields, there are a lot of optimization problems. Among them, most of the optimization problems have a variety of constraints. Evolutionary optimization method has been widely used to solve a variety of optimization problems, but the use of this method to solve constrained optimization problems, it is necessary to melt the necessary constraint processing technology [1]. In recent years, human penalty function technology is used in evolutionary optimization methods to solve constrained optimization problems. However, this method has a reasonable penalty factor setting problem: the penalty factor is too large, although it can make the evolutionary population converge quickly to the feasible domain of the problem, but neglects the value of the population to be of no value. By using the feasible solution, it is difficult to find the optimal solution of the problem; on the contrary, if the penalty factor is set too small, then the evolutionary population will stay in the unfeasible region [2][3], and the other way to deal with the constraints is that it is difficult to get the feasible solution of the problem. The problem of optimization is transformed into a multi-objective optimization problem without constraints, which is solved by using the existing unconstrained optimization methods. However, most methods require the infeasible solution to be inferior to the feasible solution when comparing the performance of the individual, so it is difficult to fill in the valuable information provided by the infeasible solution at the early stage of evolution. In addition, the effectiveness of the method also depends on the specific multi-objective optimization method [4][5].

In order to solve constrained multi-objective optimization problems, many scholars propose improvement methods. M.

Wang [6] proposed a partial order relation in the improved algorithm and used Cauchy distribution to do crossover operations. Then, according to the partial order relation, the individuals are sorted and generated.. W. Zhang [7] proposed an improved NSGA-II algorithm based on differential evolution mechanism. Z. Han [8] proposed a multi-objective dynamic adaptive differential evolution algorithm (MODADE) to improve the search efficiency of general differential evolution operations. I. O. Essiet [9] proposed that the combination of reference points and adaptive crossover rate can improve the performance of NSGA- II.

## II. NSGA-II ALGORITHM

NSGA-II is proposed by Deb et al [10]. This algorithm is one of the most excellent multi-objective optimization evolutionary algorithms, and has been widely applied [11-13]. The core of NSGA-II evolutionary algorithm lies in two aspects: the fast non-dominated sorting and elitist selection strategy for individuals. The fast non-dominated sorting is based on non dominated rank and crowding distance index. The Pareto control relation between the optimal energy functions determines the rank of the non-dominated function. Take a multi-objective optimization problem with two performance indicators as an example. Pareto domination is defined as follows: for the solution vector  $x_1$  and  $x_2$  in the feasible field,  $x_1$  is called Pareto domination or  $x_1$  dominates  $x_2$  (donate  $x_1 \not\phi x_2$ ), when the following condition holds:

$$\begin{aligned} f_i(x_1) &\leq f_i(x_2), \forall i \in (1,2) \wedge \\ f_j(x_1) &\leq f_j(x_2), \exists j \in (1,2) \end{aligned} \quad (1)$$

The crowd distance index of each individual is a representation of the distribution uniformity of the current population.

The evolution of the NSGA-II algorithm is shown in Figure I.

Individual choice can be described as follows: first, take the second generation evolution as an example, the whole individual comes from two parts: Parent individuals (Pt) and Offspring individuals (Qt) from the last generation; secondly, calculate the non-dominated rank F and crowding distance index for each individual in the new population of (Pt, Qt).

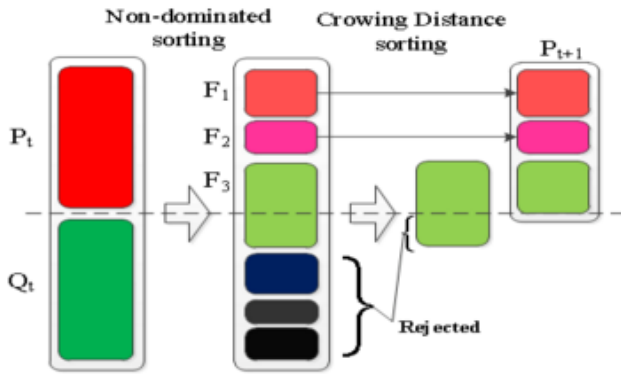


FIGURE 1. EVOLUTION PROCESS OF NSGA-II ALGORITHM [10]

According to the level of each individual (small rank is priority), excellent individuals are selected to the elite group. For individuals with the same rank, individuals with crowding distance index and large crowding distance have priority. Then, the TTH generation evolved until the population size reached  $N$ , forming a fresh parent population  $P_{t+1}$ .

### III. CONSTRAINT OPTIMIZATION PROBLEM AND ITS RELATED DESCRIPTION

Generally speaking, a constrained optimization problem can be described as follows:

$$\begin{cases} \min f(x), x \in D \\ g_i(x) \leq 0, i = 1, 2, \Lambda, q \\ h(x) = 0, j = q + 1, \Lambda, m \end{cases} \quad (2)$$

In the formula,  $f(x)$  is the objective function, and  $x$  is a decision variable.  $g_i(x) \leq 0, i = 1, 2, \Lambda, q$  is the inequality constraint condition of the problem,  $D = \{x \in R^n | l_i \leq x_i \leq u_i, l_i, u_i \in R, i = 1, 2, \Lambda, n\}$  is a search space, all the individual solutions satisfying the constraint condition in the search space  $D$  form feasible solutions.  $h(x) = 0, j = q + 1, \Lambda, m$  is an equation constraint condition.  $|h_j(x)| - \delta \leq 0$ ,  $\delta$  is an equality constraint that violates function thresholds and is generally set to a smaller positive number. In the evolutionary process. Define

$$G_i(x) = \begin{cases} \max\{0, g_i(x)\}, 1 \leq i \leq q \\ \max\{0, h_i(x) - |\delta|\}, q + 1 \leq i \leq m \end{cases} \quad (3)$$

It represents the violation degree of a solution  $x$  to the  $i$  constraint in a group, and reflects the distance order of  $\delta$  from the feasible domain.

### IV. MULTI-OBJECTIVE CONSTRAINED OPTIMIZATION

Based on the traditional NSGA algorithm, this paper proposes an improved algorithm. In the processing of the constrained optimization problem of NSGA algorithm, the constraint conditions are converted to the target of optimization. In the search process, the comparison rule of Deb is used to compare the feasible solution with the infeasible solution. In practical applications, the real optimal solution of most constrained optimization problems is often located in the vicinity of the constrained boundary, and the probability of the objective function of the unfeasible solution located in the constrained boundary is superior to that of the feasible solution of the target function. Therefore, the utilization of these highly infeasible solutions can enhance the search speed closer to the feasible domain.

Considering the difference between feasible solution and infeasible solution, it is proposed that genetic operators can be selected from every evolutionary number of generations with better feasible solution set and infeasible solution set. The concept of evolutionary optimization is proposed: for a two objective optimization problem, assuming that the parent individual  $P_1, P_2$ , sub-generation individual  $C_1, C_2$ , parent individual and progeny corresponding target function values are  $f_p^1, f_p^2, f_c^1, f_c^2$  respectively. Two of the nine possible relationships for individuals to compare, the first four cases belong to the evolutionary preferred range.

$$\begin{aligned} & \left\{ \begin{matrix} f_c^1 < f_p^1 \\ f_c^2 < f_p^2 \end{matrix} \right\}, \left\{ \begin{matrix} f_c^1 < f_p^1 \\ f_c^2 = f_p^2 \end{matrix} \right\}, \left\{ \begin{matrix} f_c^1 = f_p^1 \\ f_c^2 < f_p^2 \end{matrix} \right\}, \left\{ \begin{matrix} f_c^1 = f_p^1 \\ f_c^2 = f_p^2 \end{matrix} \right\} \\ & \left\{ \begin{matrix} f_c^1 > f_p^1 \\ f_c^2 > f_p^2 \end{matrix} \right\}, \left\{ \begin{matrix} f_c^1 < f_p^1 \\ f_c^2 > f_p^2 \end{matrix} \right\}, \left\{ \begin{matrix} f_c^1 > f_p^1 \\ f_c^2 < f_p^2 \end{matrix} \right\}, \left\{ \begin{matrix} f_c^1 > f_p^1 \\ f_c^2 = f_p^2 \end{matrix} \right\} \\ & \left\{ \begin{matrix} f_c^1 = f_p^1 \\ f_c^2 > f_p^2 \end{matrix} \right\} \end{aligned} \quad (4)$$

Genetic manipulation is effective to compare offspring and parental individuals.

$$c_1 \propto p_1, c_1 \propto p_2, c_2 \propto p_1, c_2 \propto p_2 \quad (5)$$

It is desirable to have at least one generation of individuals. This operation can ignore the poorly performing evolutionary process, exclude individuals with poor characters and ensure that the search is faster and more efficient to get closer to the optimal solution area, and the optimization performance of the algorithm is improved. After generation, the unfeasible solution and feasible solution are executed and reorganized, and an adaptive adjustment mechanism is set up for this link. Considering that the search is close to the feasible domain and the optimal solution direction in the global optimization process, the number of feasible solutions is gradually increased as far as the whole process of evolution is concerned. If the

genetic operation that executes the feasible solution and the infeasible solution too much in the later stage of evolution, it may counteract the effect of the algorithm in the feasible domain. Therefore, we should gradually reduce the frequency of genetic operations between infeasible solutions and feasible solutions in search process. In this paper, the evolutionary algebra of the feasible solution and the intersecting link of the infeasible solution are adaptively adjusted in the process of executing the genetic operation between the feasible solution and the infeasible solution, that is, when the evolutionary algebra is  $\delta$ , the cross variation of the infeasible solution and the feasible solution is performed.

$$\begin{cases} K = \text{round}(k) \\ k = \{k \leq T | k = e', t \in [0, 1, 0.1, 0.2, 0.3, \Lambda]\} \end{cases} \quad (6)$$

In the formula, K is the evolutionary number of the feasible and infeasible solutions to perform crossover and mutation operations. T is the total population evolution algebra.

In order to effectively retain the dominant individual in the process of population evolution to better play its guiding role in the direction of evolution, this paper preserves the first frontier feasible solution archive as an advantage set in the optimization process, and ensures that the replacement link is effective in the process of replacing the dominant set. Compared with individuals  $a$  and elite individuals  $b$ ,  $b$  has better performance than  $a$  and individual substitution. In view of the serious overlap of dominant individuals in the feasible domain of search, this paper proposes half performing marginal mutation operation on the dominant individual size. In the process of alienation from  $x$  to  $x'$ , if the value range of point C is  $[U_{\min}, U_{\max}]$ , Then  $x'_k$  can be obtained by the following:

$$x'_k = \begin{cases} U_{\min}^k, \text{rand}(0,1) = 0 \\ U_{\max}^k, \text{rand}(0,1) = 1 \end{cases} \quad (7)$$

$\text{rand}(0,1)$  indicates that 0 or 1 are selected with the same probability. If the range of decision variables is large and unconstrained, marginal variation will probably not improve the effectiveness of the algorithm. For the case where the optimal solution is at the boundary of the feasible domain, the marginal mutation operation focuses on the global. The key point is that the individual solution is still located in the feasible domain of the target space after the mutation, and the global searching ability of the algorithm is improved and the performance of the algorithm is improved.

The process steps for improving the algorithm NSGA-II are as follows:

- 1) Set algorithm parameters and initialize population.
- 2) To distinguish the population set of feasible solutions and infeasible solution set.

3) Non dominated sorting, crowding distance calculation and feasible selection operators are used to determine the first parent.

4) Perform advantage set archiving and replace parental individuals.

5) Performing genetic manipulation

6) To perform intermediate selection operations for parent-child mixed sets to determine the next parent generation and determine whether evolution reaches the maximum evolutionary algebra, if it is the end of the algorithm; otherwise, jump to step 7)

7) To determine whether evolution can reach the conditions to carry out genetic operations between infeasible solutions and feasible solutions, if so, turn step 8); otherwise, turn to step 4).

8) Performing genetic operations on infeasible solutions and feasible solutions.

9) The next generation of feasible solutions update the next parent, and the infeasible solution is retained as the next parent, then, turn to step 3).

## V. EXPERIMENTS

This article tests the typical test functions (ZDT1-2) listed in [10]. Using NSGA-II algorithm and INSGA-II algorithm as comparison algorithms. Converting equality constraint  $h(x) = 0$  in optimization function to  $|h(x)| - \delta \leq 0$  inequality constraint,  $\delta = 0.0001$ , Population size is 500, Evolutionary algebra is 200, The crossover probability is 0.9, The probability of mutation is 0.1. Experimental results shown in Figure II, Figure III.

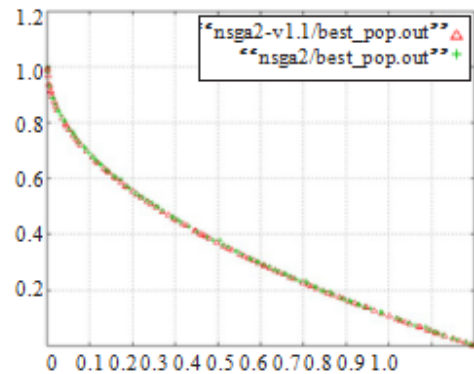


FIGURE II. COMPARISON OF EXPERIMENTAL RESULTS OF TWO ALGORITHMS ON ZDT1

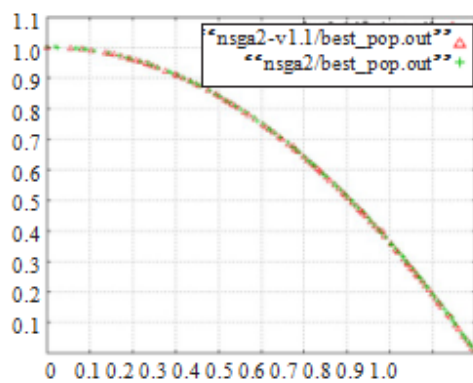


FIGURE III. COMPARISON OF EXPERIMENTAL RESULTS OF TWO ALGORITHMS ON ZDT2

From the experimental results, it can be seen that both algorithms can obtain a Pareto frontier with better convergence, and the Pareto frontier is very similar to the real frontier, however, the Pareto frontier obtained by the improved algorithm is uniformly distributed, the Pareto frontier obtained by the NSGA-II algorithm has local leakage, and the uniformity is slightly weaker. On the other hand, the number of solutions in the Pareto frontier obtained by the improved algorithm is more than the number of Pareto frontier solutions obtained by the original algorithm.

## VI. CONCLUSION

In this paper, based on the classical NSGA-II algorithm, an improved algorithm NSGA-II algorithm for solving constrained multi-objective optimization problems is proposed. Based on the NSGA-II algorithm, this algorithm performs genetic operations on infeasible solutions and feasible solutions every certain evolutionary algebra and adaptively adjusts the frequency of this genetic operation. Set restrictions to ensure the validity of genetic operations. Solving the problem of overlap of late dominant individuals. Experimental results show that the algorithm has superiority.

## REFERENCES

- [1] B. Tessema and G. G. Yen, "An Adaptive Penalty Formulation for Constrained Evolutionary Optimization," in *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 39, no. 3, pp. 565-578, May 2009.
- [2] Correia A, Mestre P, Serodio C, et al. Adaptive Penalty and Barrier function based on Fuzzy Logic[J]. *Expert Systems with Applications*, 2015, 42(19):6777-6783.
- [3] Lemonge A C C, Barbosa H J C, Bernardino H S. Variants of an adaptive penalty scheme for steady-state genetic algorithms in engineering optimization[J]. *Engineering Computations*, 2015, 32(8):2182-2215.
- [4] Deb K. An efficient constraint handling method for genetic algorithms[J]. *Computer Methods in Applied Mechanics & Engineering*, 2000, 186(2):311-338.
- [5] Sarker R A, Elsayed S M, Ray T. Differential Evolution With Dynamic Parameters Selection for Optimization Problems[J]. *IEEE Transactions on Evolutionary Computation*, 2014, 18(5):689-707.
- [6] Wang M, Dai G, Hu H. Improved NSGA-II Algorithm for Optimization of Constrained Functions[C]// *International Conference on Machine Vision and Human-Machine Interface*. IEEE, 2010:673-675.
- [7] Zhang W, Zhang J L. Improved NSGA-II algorithm based on differential evolution mechanism[C]// *Chinese Control Conference*. 2017:4334-4338.
- [8] Han Z, Wang S, Dong X, et al. Improved NSGA-II algorithm for multi-objective scheduling problem in hybrid flow shop[C]// *International Conference on Modelling, Identification and Control*. IEEE, 2018:5-10.
- [9] I. O. Essiet, Y. Sun and Z. Wang, "Enhanced NSGA based on adaptive crossover rate and reference points," 2017 2nd International Conference on Robotics and Automation Engineering (ICRAE), Shanghai, 2017, pp. 295-300.
- [10] Deb K, Pratap A, Agarwal S, et al. A fast and elitist multiobjective genetic algorithm: NSGA-II[J]. *IEEE Transactions on Evolutionary Computation*, 2002, 6(2):182-197.
- [11] Zeinalzadeh A, Mohammadi Y, Moradi M H. Optimal multi objective placement and sizing of multiple DGs and shunt capacitor banks simultaneously considering load uncertainty via MOPSO approach[J]. *International Journal of Electrical Power & Energy Systems*, 2015, 67:336-349.
- [12] Li Y, Lu X, Kar N C. Rule-Based Control Strategy With Novel Parameters Optimization Using NSGA-II for Power-Split PHEV Operation Cost Minimization[J]. *IEEE Transactions on Vehicular Technology*, 2014, 63(7):3051-3061.
- [13] Zamanifar M, Fani B, Golshan M E H, et al. Dynamic modeling and optimal control of DFIG wind energy systems using DFT and NSGA-II[J]. *Electric Power Systems Research*, 2014, 108(3):50-58.