A New Particle Swarm Optimization Algorithm for Solving Constraint and Mixed Variables Optimization Problem

Weigang Wang
Mechanical Science and Engineering College, Daqing Petroleum Institute, Daqing, China
wwgnhm@yahoo.com.cn

Hongmei Ni
Computer and Information Technology College, Daqing Petroleum Institute, Daqing, China,nhm257@163.com

Abstract

Many engineering optimization problems frequently encounter mixed variables and nonlinear constraints, which add considerably to the solution complexity. Very few of the existing methods can yield a globally optimal solution when the objective functions are non-convex and non-differentiable. We developed a new particle swarm optimization (PSO) algorithm. The algorithm introduced a mechanism of simulated annealing (SA), crossover and mutation operator. It may improve the evolutionary rate and precision of the algorithm. We put forward a method of stochastic approximation, in order to realize the transformation from continuous variable to discrete variable. For handling constraints, we used death penalty function method. Based on engineering design problem, computational result was better than the other solutions reported in the literature. Therefore, the new algorithm is feasible, and its accuracy and robustness are obviously superior to the other algorithms.

1. Introduction

The optimization design in engineering, such as pressure containers, shell-and-tube heat exchanger and Marine engineering structure optimization design, etc, mostly belongs to nonlinear constraints and the mixed-discrete variable optimization design. And its design parameters include continuous variables and discrete variables (integer variables are regarded as special discrete variables). Due to the discrete variables, the traditional optimization design theory and the algorithm used in the continuous variables are not suitable for discrete variable structure optimization problem, and the optimization problems become very difficult. The optimization design of discrete variables is often classified into the following three algorithms [1]: the exact algorithm, the approximate algorithm and heuristic algorithm. Genetic algorithm, the hybrid genetic algorithm, the PSO algorithm and simulated annealing method are the latest design method.

The PSO algorithm [2] is originated from the observation and research for the predator behavior of bird flock, which is firstly put forward by Kennedy and Eberhart in 1995. Because the PSO algorithm is simple, fast convergence rate, and good robustness etc, it is widely used in many areas. Currently, the research and application on discrete PSO algorithm applied in engineering field are in the early stage. Many scholars put forward their own new algorithm [3-6]. Due to the limitations of various algorithms and the features of discrete variable, carrying on the study of high precision, high efficiency and stable convergence algorithms has very important practical significance.

A new PSO algorithm was brought forward. We introduced simulated annealing technique in the algorithm. This algorithm, based on constraints and mixed variables treatment strategies, can handle with simultaneously nonlinear constraints, continuous and discrete variables. Thus, we may carry on easily the engineering optimization design.

2. Basic theory

2.1. The transformation of the mathematical model

Engineering optimization design problems generally may be summed up the following mathematics model:

$$\begin{aligned} &\min f(\mathrm{X}) \\ &\text{s.t} \quad h_j(\mathrm{X}) = 0 \qquad j = 1, 2, \cdots, p \\ &\qquad g_j(\mathbf{X}) \le 0 \qquad j = p+1, p+2, \cdots, m \\ &\qquad X_i' \le X_i \le X_i'' \qquad i = 1, 2, \cdots, n \end{aligned} \tag{1}$$

Where, $\mathrm{X} = \{X_1, X_2, \cdots, X_2\}^T = \{X^c, X^d\}^T$, X^c is the subset of continuous variables, X^d is the subset of discrete variables, $f(\mathrm{X})$ is the objective function being optimized, the latter three formulas denote

978-1-4244-7703-6/10/$26.00 ©2010 IEEE

respectively the equality constraint, inequality constraint and the numeric area of variables.

2.2. A new PSO algorithm thought

In new PSO algorithm, the basic PSO algorithm flow is acted as the main process. Inertia weight [7] is decreased in the linear, firstly carrying on global search, and then strengthening the capacity of local search. At the same time, using crossover [8] operation and Gaussian mutation[9], in order to introduce simulated annealing [10] into them, so that group was further optimized.

Kennedy and Eberhart [2] have given the overview of standard PSO algorithm. We no longer go into details here. In order to describe conveniently, only iterative equation about the algorithm is listed below.

$$v_i^{t+1} = \omega v_i^t + c_1 r_1 (p_i^t - x_i^t) + c_2 r_2 (p_g^t - x_i^t) \quad (2)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3)$$

Where, v_i^t denotes velocity vector of particle i at iteration t, and $v_i \in [-v_{i\max}, v_{i\max}]$. x_i^t denotes the current position of particle i at iteration t. p_i^t denotes the best position attained so far by particle i at iteration t. p_g^t denotes the global best position of the group at iteration t. $i = 1, 2, \cdots, N$, N denotes the size of the swarm. ω denotes inertia weight; c_1, c_2 are two positive constants, called cognitive and social parameter, respectively. r_1, r_2 are random numbers, uniformly distributed between 0 and 1.

Note that particle position x_i^t, particle velocity v_i^t, particle best position p_i^t and the swarm best experience p_g^t are all vectors of dimension *d*.

According to the crossover probability P_c preset, some particles are chosen to cross between one particle and the other from the particle swarm generated in each iteration. Then the offspring particles produced replace the parent particle to form a new particle swarm.

The update formula of particle position is as follows:

$$x_1' = p * x_1 + (1-p) * x_2 \text{ and } x_2' = p * x_2 + (1-p) * x_1 \quad (4)$$

The update formula of particle velocity is as follows:

$$v_1' = \frac{v_1 + v_2}{|v_1' + v_2'|}|v_1| \quad \text{and} \quad v_2' = \frac{v_1 + v_2}{|v_1' + v_2'|}|v_2| \quad (5)$$

Where, x_1, x_2, v_1, v_2 denote the position and velocity of two parent particles; x_1', x_2', v_1', v_2' denote the position and velocity of two offspring particles; p is uniformly distributed random vector.

According to a mutation probability P_m preset, some particles are chosen to mutate with Gaussian for the particle swarm generated in crossover operation. Then the new particles after being mutated replace the original particle. Mutation formula is as follows:

$$x_j^k = (x_{\max}^k + x_{\min}^k)/2 + (x_{\max}^k - x_{\min}^k)/2 \times (\sum_{i=1}^{12} r_i - 6) \quad (6)$$

Where, j denotes No. j particle, k denotes the k-dimension, $x_{\max}^k$, $x_{\min}^k$ denotes the maximum value and minimum value in the position, r is random number uniformly distributed between 0 and 1.

The inertia weight decreases linearly from a maximum to a minimum with the increases of iteration number. The algorithm firstly search from global, then search further from local. The specific weight formula is as follows:

$$\omega = \omega_{\max} - (\omega_{\max} - \omega_{\min}) \times t / T_{\max} \quad (7)$$

Where, $\omega_{\max}$ is the maximum of ω; $\omega_{\min}$ is the minimum of ω; t is the current iteration number; $T_{\max}$ is the maximum iteration number.

2.3. New PSO algorithm flow

Step1. Initialize parameters: c_1, c_2; $\omega_{\min}$; $\omega_{\max}$; N; $T_{\max}$; P_c; P_m; v_i^1; the initial annealing temperature T; the temperature cooling coefficient C.

Step2. Calculate the fitness value of each particle, find the extreme value p_i for each individual particle and global extremum p_g of particle swarm, and thus determine the two-optimization directions of each particle.

Step3. Compare the fitness of each particle with the individual extreme value p_i. If current value is better than p_i, then set the current particle as the p_i.

Step4. Compare the fitness of each particle with the global extreme value p_g. If current value is better than p_g, then set the current particle as the p_g.

Step5. Update each particle's position and velocity according to equation (2) and (3), and then generate new groups.

Step6. Choose particles from the population produced in Step 5 according to the crossover probability p_c as sub-populations. Namely, use simulated annealing mechanism to accept the offspring after being crossed. Then optimize and adjust it to produce a new population.

Select randomly the individual x_j and x_k from the sub-populations. Cross according to equation (4), (5).

Produce the two new individuals x_j' and x_k'. Calculate the fitness function value $f(x_j)$, $f(x_k)$, $f(x_j')$ and $f(x_k')$.

If $\min\{1, \exp(-(f(x_j) - f(x_j'))/T)\} > rand$, act x_j' as the new individual;

If $\min\{1, \exp(-(f(x_k) - f(x_k'))/T)\} > rand$, act x_k' as the new individual.

Where, *rand* is the random number, uniformly distributed between 0 and 1.

Step7. Choose particles from the new population after being crossed to produce sub-populations according to the mutation probability P_m. Namely, use simulated annealing mechanism to accept the offspring after being mutated, and then produce a new population:

Select randomly the individual x_j from the sub-populations. Cross according to equation (6), and then produce new individuals x_j'. Calculate the fitness function $f(x_j)$, $f(x_j')$.

If $\min\{1, \exp(-(f(x_j) - f(x_j'))/T)\} > rand$, act x_j' as the new individual.

Step8. If the current best individual meets the convergence condition, the evolution process is complete. Then return to the global optimal solution.

Step9. If the number of iteration has not reached the maximum preset number, then modify the annealing temperature of the population, namely, $T = CT$, go to step 2.

The offspring after crossover and mutation has been optimized to adjust through the simulated annealing idea from step 6 to step 8. Namely, in early phase, the acceptance probability for offspring is larger, in order to increase the diversity of population, and avoid getting algorithm into local optimum. Nevertheless, in late phase, the probability of acceptance for the offspring gradually becomes smaller with the annealing temperature decreasing gradually. Moreover, the probability for non-optimum solution is reduced. So the speed of evolution algorithm is enhanced.

2.4. Constraint processing strategy

When dealing with constraints, these particles, which don't meet constraint conditions, are all received with total probability by taking examples from penalty function. Namely, all particles without meeting the constraints of the previous generation are also involved in generating the next generation of evolution. However, their objective functions are given with a large constant value (assuming that search the global minimum in the paper). The expression is as follows:

$$f(X_i) = \begin{cases} f(X_i) & \text{satisfaction} \\ c_{\max} & \text{notsatisfaction} \end{cases} \tag{8}$$

Where, $f(X_i)$ is $X_i (i = 1, 2, 3, \cdots, n)$ corresponding to the objective function values; $c_{\max}$ is made as a large constant, the paper is as 9999999.

2.5. Treatment strategy with mixed variables

Aiming at the characteristics of the PSO algorithm, a stochastic approximation method is brought forward to deal with mixed variables. The idea is as follows: Integer variable M is looked as continuous variable to search. Select randomly one of two integers, $\text{floor}(r)$ and $\text{ceil}(r)$ adjacent to continuous variable values r, as the value of integer variable M. But the choosing probability is inversely proportional to the distance with them to r. If such a deal is thought as the INTR operation, then:

$$\mathrm{M} = \mathrm{INTR}(r) = \begin{cases} \text{floor}(r) & \text{if rand} > r - \text{floor}(r) \\ \text{ceil}(r) & \text{otherwise} \end{cases} \tag{9}$$

Where, r denotes a continuous variable; rand is random number, uniformly distributed between 0 and 1; $\text{floor}(r)$, $\text{ceil}(r)$ denotes respectively rounding the elements of r to the nearest integers less than or equal to r, greater than or equal to r.

For the other discrete variables besides integer variables, their index number is taken as integer variable. Then we deal with them according to above method. It should be point out that all dimensions corresponding to other discrete variables are actually the index number of these variables. However, it needs to substitute the discrete values for index number in assessing the objective function and the constraint violation degree.

3. Project example and result analysis

Design a container with the least material. The design variables are:

$$X = [x_1, x_2, x_3, x_4]^{\mathrm{T}} = [R, L, t_s, t_h]^{\mathrm{T}}$$

Where, R, L are respectively tube length and radius of container, being the integer multiple of 1; t_s、t_h are respectively head thickness and wall thickness, being the integer multiple of 0.0625.

Through derivation and arrangement, the mathematical model is as follows:

$$
\begin{aligned}
&\min f(X) = 0.6224x_1x_2x_3 + 1.7781x_1^2x_4 \\
&\qquad + 3.1661x_2x_3^2 + 19.84x_1x_3^2 \\
&\text{s.t } 10 \le x_1, x_2 \le 200 \\
&\quad 0.0625 \le x_3, x_4 \le 6.1875 \\
&\quad g_1(x) = 0.0193x_1 / x_3 - 1 \le 0 \\
&\quad g_2(x) = 0.00954x_1 / x_4 - 1 \le 0 \\
&\quad g_3(x) = x_2 / 240 - 1 \le 0 \\
&\quad g_4(x) = \frac{(1296000 - 0.75\pi x_1^3)}{\pi x_1^2 x_2} - 1 \le 0
\end{aligned} \tag{10}
$$

The new PSO algorithm is applied to optimize above problem. We actualize the algorithm in MATLAB. The initialization parameters are as follows: $N = 30$, $\omega_{\min} = 0.6$, $\omega_{\max} = 0.9$, $P_c = 0.5$, $P_m = 0.05$, $C = 0.8$, $T = 100000$, $T_{\max} = 1000$, $v_{\max} = 0.5$, $c_1 = c_2 = 1.50$.

Based on the above parameters, carry on optimization design for the container with mixed variables. The result of the paper and the results of literature [11-13] are listed in Table 1.

Table 1. Compare the paper result with the literature results

Optimal results	KANNAN[11]	CARLOS[12]	HE[13]	New PSO
x_1	58.29	40.32	42.10	44.00
x_2	43.36	200.00	176.64	163.00
x_3	1.125	0.8125	0.8125	0.8125
x_4	0.625	0.4375	0.4375	0.4375
$f(X)$	7198.200	6288.745	6059.710	6049.910

By analyzing above results, it is clear that the algorithm proposed in the paper is feasible, and is superior to other algorithms.

4. Conclusion

In this paper, the mechanism of simulated annealing is involved into the original PSO algorithm with crossover and Gaussian mutation. In addition, inertia weight is decreased in the linear. The new PSO algorithm can improve the ability of seeking the global excellent value, the convergence speed and accuracy. We put forward two kind of treatment strategy aiming at the nonlinear constraints and mixed variables problem. Project example has proved that the new PSO algorithm cannot only solve the nonlinear constrained and mixed variables problems in engineering, but also has the faster evolutionary velocity, the higher convergence accuracy and the stronger global optimization ability.

5. References

[1] YAN Fei, CHEN Lei, “Development and trend for optimum design method of discrete variable structures”, Shangxi Architecture, 2007, 33(7), pp. 101-103.

[2] Kennedy J, Eberhart R, “Particle swarm optimization”, In: IEEE Int’l Conf on Neural Networks, Perth, Australia, 1995, pp. 1942-1948.

[3] JUAN J, Particle swarm optimization applications in power system engineering, Puerto Rico, University of Puerto Rico, 2004.

[4] TAYAL M, Particle swarm optimization for mechanical design, Texas, University of Texas, 2003.

[5] LI Xiang-yong, TIAN Peng, KONG Min, “A New Particle Swarm Optimization for Solving Constrained Optimization Problems”, Journal of Systems & Management, 2007, 16(2), pp. 120-129, 134.

[6] WANG Jin-hua, YIN Ze-yong, “Solving constrained discrete optimization problem with Particle Swarm Optimizer”, Computer Engineering and Applications, 2008, 44(3), pp. 242-244.

[7] YUAN Peng, CHANG Jiang, ZHU Bing, LI Bin, “ Applying the inertia weight model of particle swarm optimization in reservoir flood control operation”, Journal of Sichuan University(Engineering Science Edition), 2006, 38(5) , pp. 54~57.

[8] Angeline P J, “Evolutionary optimization versus particle swarm optimization: Philosophy and performance differences”, In: Evolutionary programming Ⅶ, 1998, pp. 601-610.

[9] Natsuki Higashi, Hitoshi Iba, “Particle swarm optimization with Gaussian mutation”, Proceedings of the IEEE Swarm Intelligence Symposium 2003, Indianapolis, Indiana, USA, 2003, pp. 72-79.

[10] Kang Lishan , Non-numerical parallel algorithms (Volume 1) - simulated annealing algorithm, Science Press, Beijing, 1997.

[11] KANNAN B K, KRAMER S R, “An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its application to mechanical design”, Journal of Mechanical Design, Trans. ASME, 1994, 116, pp. 318-320.

[12] CARLOS A, “Use of self-adaptive Penalty approach for engineering optimization Problems”, Computer in Industry, 2000, 41, pp. 113-127.

[13] HE S, “An improved Particle swarm optimizer for mechanical design optimization Problems”, Engineering Optimization, 2004, 35(5), pp. 585-605.