Reducer optimization design based on chaotic particle swarm optimization (CPSO)

Jiong Cai*

Panzhihua University, Panzhihua, Sichuan, 617000, China

Received 1 March 2014, www.cmnt.lv

Abstract

For the optimization design of two-stage gear reducer, an optimization mathematical model is built in this work to determine the objective functions and constraints. And chaotic particle swarm optimization (CPSO) is utilized to optimize these functions and constraints. Algorithm simulation is carried out based on CSPO algorithm steps, and the results are compared with particle swarm optimization (PSO). Simulation indicates that CSPO can optimize the results of PSO and achieve faster convergence rate.

Keywords: chaos particle swarm algorithm, optimization design, mathematical model, reducer

1 Introduction

As a conventional mechanical device, reducer is widely utilized in heavy machinery including mining machinery, construction machinery and transportation machinery. Its main role is to reduce motor speed or increase motor torque, so reducer optimization design has considerable theoretical and practical value.

2 Particle swarm algorithm

2.1 ALGORITHM PRINCIPLE

Particle swarm [4-7] optimization algorithm simulates predatory behavior of birds, make groups to achieve purpose through the collective cooperation among birds. In particle swarm optimization (PSO) algorithm, a bird is called "particle"; solving group is equivalently bird fauna; the migration from one location to another is equivalently evolution of the population; "good news" is equivalently the local optimization of population; food sources is equivalently the global optimal solution of population. In particle swarm model, the search space is D-dimension; and the total number of particles is n. Each optimization goal is the state of "particles" in the search space, including speed and position. Secondly every particle has a fitness value decided by the optimization function, and also a speed determines their flight direction and location. According to flying experience of oneself and the companions, the particles adjust dynamically the status, that is to say, update oneself through updating two positions. One is the individual best position id p found by particles themselves; another is the global best position found by entire population.

Particle swarm algorithm in operation process randomly generates an initial population and gives each particle a random speed, then update the particle speed and position according to

$$v_{id} = wv_{id} + c_1 r_1 \left(p_{id} - x_{id} \right) + c_2 r_2 \left(p_{gd} - x_{id} \right), \quad (1)$$

$$x_{id} = x_{id} + v_{id} , \quad (2)$$

$$v_{id} \left(t+1 \right) = v_{id} \left(t \right) + c_1 rand_1 \left(p_{pid} - x_{id} \left(t \right) \right) + c_2 rand_2 \left(p_{gd} - x_{id} \left(t \right) \right). \quad (3)$$

Randomly generate the initial position and velocity of particle swarm, and then execute iteration according to Equations (1-3) until satisfactory solution was found. Convergence rate of particle swarm algorithm is fast; it is easy to implement and the number of the parameter needed to adjust is less. It has become a new hotspot of study in intelligent optimization evolutionary computation field. Its advantages aroused the attention of academic circles, such as simple implementation, high accuracy, fast convergence rate, strong approximation ability, and the algorithm shows its advantages in the solving actual problem.

3 Chaos particle swarm optimization

For the problem of local optimum in particle swarm algorithm, the work presents the chaos theory for improvement of PSO algorithm, and the process is as follows [8-10]:

1) In chaos initialization, supposing the variable to be optimized is D-dimensional, a D-dimensional vector $z_1 = \left[z_{11}, z_{12}, ..., z_{1D} \right]$ is randomly generated, and each component is within the range of $[0,1]$. Then *M* components are obtained according to the logistic equation [8], $z_1, z_2, ..., z_M$.

$$z_{n+1} = \mu z_n (1 - z_n), n = 0,1,2,...; 0 < z_n < 1; \mu \in [0,4] . \quad (4)$$

***Corresponding author** e-mail: jiongcaipzh@163.com

The chaotic interval will be mapped to the range of variable according to Equation (5).

$$x_{ij} = a_j + (b_j - a_j) z_{ij} , \quad (5)$$

where b_j, a_j are the upper and lower limits of optimized variable, respectively.

2) The fitness value of each particle is calculated using objective function. The *N* particle swarms with better performance are chosen as the initial solution from the *M* initial swarms, randomly generating particle velocity.

3) The initial individual and global extreme of particles are set: the current position of each particle is defined as individual extreme P_i, thereby calculating the corresponding fitness value of each individual extreme based on objective function; the position of particle with the optimal value is defined as global optimum P_g.

4) The flight speed and position of particles are updated according to velocity-position updating formula.

5) Chaos optimization is conducted on optimal position P_g: firstly, the optimal position is mapped to the defined domain of logistic equation [11] using Equation (6). Then, according to logistic equation, the iteration process generates *m* chaotic variable sequences. Finally, these sequences are mapped to the value interval of optimization variable, obtaining *m* particles. Fitness values of each particle are calculated for the optimal solution *p'*.

$$z_9 = \frac{P_g - a_i}{b_i - a_i} . \quad (6)$$

6) The current position of any particle in the swarm is substituted by *p'*.

7) The algorithm will return to Step 4 until the termination condition of particle swarm is fulfilled. Then it will stop calculating and output the results.

4 Reducer optimization design model

In this work, two-grade gear reducer is the object of design study, and the mechanism chart is shown in Figure 1 [12].

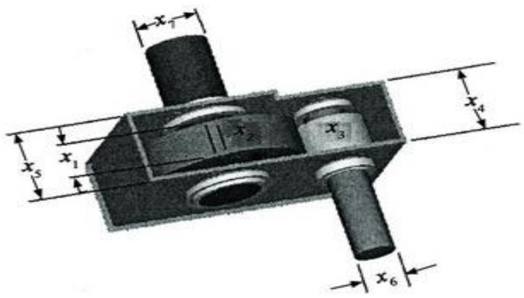


FIGURE 1 Reducer mechanism chart

This problem of design optimization has seven design variables, gear face width x_1, tooth mold x_2, tooth number of small gear x_3, bearing spacing of axis 1 x_4, bearing spacing of axis 2 x_5, the diameter of axis 1 x_6 and the diameter of axis 2 x_7. The ranges of these variables are as below:

$$\begin{aligned} &2.6 \le x_1 \le 3.6, 0.7 \le x_2 \le 0.8, 17 \le x_3 \le 28, \\ &7.3 \le x_4 \le 8.3, 7.3 \le x_5 \le 8.3, 2.9 \le x_6 \le 3.9, \\ &5.0 \le x_7 \le 5.5. \end{aligned} \quad (7)$$

The smallest volume of reducers can be found utilizing objective function.

$$\begin{aligned} &\min f1(x) = 0.7854 x_1 x_2^2 (3.3333 x_3^2 + 14.933 x_3 - \\ &43.0934) - 1.508 x_1 (x_6^2 + x_7^2) + \\ &7.477 (x_6^3 + x_7^3) + 0.7854 (x_4 x_6^2 + x_5 x_7^2), \end{aligned} \quad (8)$$

$$A_1 = \left[\left(745 x_2^{-1} x_3^{-1} x_4 \right)^2 + 16.9 \times 10^6 \right]^{0.5} , \quad (9)$$

$$B_1 = 0.1 x_6^3 , \quad (10)$$

$$A_2 = \left[\left(745 x_2^{-1} x_3^{-1} x_5 \right)^2 + 157.5 \times 10^6 \right]^{0.5} , \quad (11)$$

$$B_2 = 0.1 x_7^3 . \quad (12)$$

There are 11 constraints including tooth bending stress, contact stress, axis transverse deviation, design size, etc. [13,14].

$$g_1(x) = 27 x_1^{-1} x_2^{-2} x_3^{-1} - 1 \le 0 ,$$

$$g_2(x) = 397.5 x_1^{-1} x_2^{-2} x_3^{-2} - 1 \le 0 ,$$

$$g_3(x) = 1.93 x_2^{-1} x_3^{-1} x_4^3 x_6^{-4} - 1 \le 0 ,$$

$$g_4(x) = 1.93 x_2^{-1} x_3^{-1} x_5^3 x_7^{-4} - 1 \le 0 ,$$

$$g_5(x) = x_2 x_3 - 40 \le 0 ,$$

$$g_6(x) = 5 - x_1 x_2^{-1} \le 0 ,$$

$$g_7(x) = x_1 x_2^{-1} - 12 \le 0 ,$$

$$g_8(x) = 1.9 - x_4 + 1.5 x_6 \le 0 ,$$

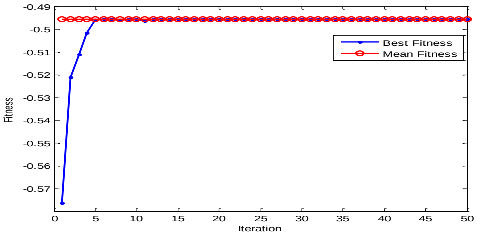
$$g_9(x) = 1.9 - x_5 + 1.5 x_7 \le 0 ,$$

$$g_{10}(x) = A_1 B_1^{-1} - 1800 \le 0 ,$$

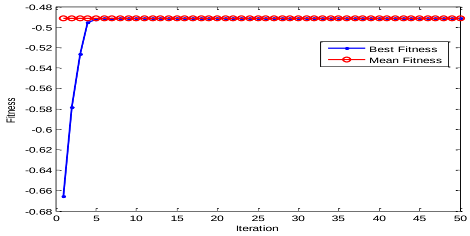
$$g_{11}(x) = A_2 B_2^{-1} - 1800 \le 0 .$$

5 Simulation

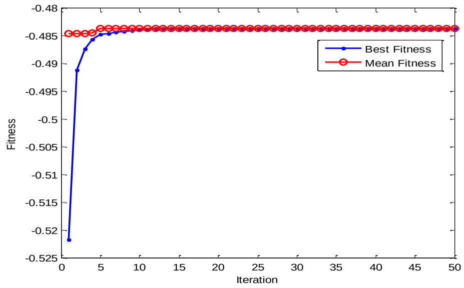
To verify the proposed algorithm, the above mathematical model is optimized and solved with the proposed method. The maximum of iteration is 50; the swarm size 20; *popmin* = –5.12; *popmax* = 5.12; *vmax* = 1; *vmin* = –1. Figure 2 shows the results of simulation.



a) Results of the first operation



b) Results of the second operation



c) Results of the third operation

FIGURE 2 CPSO simulation results

From the optimization results of CPSO algorithm, the design variables are 3.5, 0.7, 17, 7.30, 7.7153, 3.3502 and 5.2867. Figure 2 shows the convergence diagram of CPSO.

To highlight the difference between CPSO and PSO algorithms, parameters are set as follows: the maximum of iterations is 100; the swarm size 20; *popmin* = –5.12; *popmax* = 5.12; *vmax* = 1; *vmin* = –1. Figure 3 shows the convergence comparison before and after the improvement.

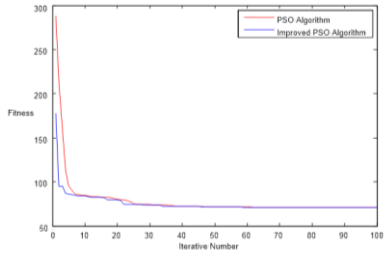


FIGURE 3 Convergence comparison of CPSO and PSO algorithm

In Figure 3, CPSO, with more stable performance, has a significantly faster convergence rate than PSO algorithm, which validates the stability and effectiveness of the proposed algorithm.

6 Conclusions

The work focuses on reducer design optimization. Chaos theory is introduced to particle swarm algorithm due to the problem of local optimum in PSO, thus proposing a CPSO algorithm for optimization design. The steps of CPSO algorithm are elaborated in detail, and the algorithm is combined with a specific case of reducer optimization design for simulation. The results indicate that the algorithm has faster convergence rate than PSO algorithm, so it is of great theoretical and practical value for engineering application.

References

[1] Donoho D L 2006 *IEEE Transactions on Information Theory* **52**(4) 1289-1306

[2] Candes E, Romberg J 2006 Quantitative robust uncertainty principles and optimally sparse decompositions *Foundations of Computational Mathematics* **6**(2) 227-54

[3] Laska J, Kirolos S, Duarte M, Ragheb T S, Baraniuk R G, Massoud Y 2007 Theory and Implementation of An Analog to Information Converter Using Random Demodulation *Proceedings of the IEEE International Symposium on Circuits and Systems 2007 (ISCAS 2007)* 1959-62

[4] Tropp J, Gilbert A 2007 *IEEE Transactions on Information Theory* **53**(12) 4655-66

[5] Donoho D, Tsaig Y 2006 Fast solution of ell-1-norm minimization problems when the solution may be sparse. Stanford University Department of Statistics Technical Report 2006-18

[6] Figueiredo M A T, Nowak R D, Wright S J 2007 Gradient projection for sparse reconstruction: Application to compressed sensing and other inverse problems *IEEE Journal of Selected Topics in Signal Processing: Special Issue on Convex Optimization Methods for Signal Processing* **1**(4) 586-98

[7] Egiazarian K, Foi A, Katkovnik V 2007 *IEEE International Conference on Image Processing 2007* I-549-I-552

[8] Duarte M, Davenport M, Takhar D, Laska J, Sun T, Kelly K, Baraniuk R 2008 Single-pixel imaging via compressive sampling *IEEE Signal Processing Magazine* **25**(2) 83-91

[9] Wakin M, Laska J, Duarte M, Baron D, Sarvotham S, Takhar D, Kelly K, Baraniuk R 2006 *2006 IEEE International Conference on Image Processing* 1273-6

[10] Wakin Wakin M, Laska J, Duarte M, Baron D, Sarvotham S, Takhar D, Kelly K, Baraniuk R 2006 *Proceedings Picture Coding Symposium (PCS)* Beijing China

[11] Takhar D, Laska J, Wakin M, Duarte M, Baron D, Sarvotham S, Kelly K, Baraniuk R 2006 A new compressive imaging camera architecture using optical-domain compression *Computational Imaging IV at SPIE Electronic Imaging* San Jose California

[12] Haupt J, Nowak R 2006 Compressive sampling vs conventional imaging *2006 IEEE International Conference on Image Processing* 1269-72

[13] Maleh R, Gilbert A, Strauss M 2007 Sparse gradient image reconstruction done faster *2007 IEEE International Conference on Image Processing* **2** II-77-II-80

Author

Jiong Cai, born in October, 1970, Nanchong, Sichuan Province, China

Current position, grades: lecturer of Panzhihua University, China.
University studies: Master's degree in Computer Science and Technology.
Scientific interests: computer networks and communications.
Publications: 12 papers.