IOSR Journal of Mechanical and Civil Engineering (IOSR-JMCE)
e-ISSN: 2278-1684,p-ISSN: 2320-334X, Volume 11, Issue 4 Ver. II (Jul- Aug. 2014), PP 120-125
www.iosrjournals.org

Optimization Design of Crane Box Girder Based on the Improved Polyclonal Selection Algorithm

Chen Chen[1], Yi Shen[1,2,*], Mingxin Yuan[1,2]
[1](*School of Mechanics and Automotive Engineering, Jiangsu University of Science and Technology, Zhangjiagang, China*)
[2](*Suzhou Institute of Technology, Jiangsu University of Science and Technology, Zhangjiagang, China*)

Abstract: *On the basis of the polyclonal selection algorithm, an improved polyclonal selection algorithm based on negative selection is put forward in this paper. Inspired by the diversity and auto-tolerance of the negative selection, the population diversity is increased by the clone deletion and clone supply to improve the polyclonal selection algorithm. Finally, the optimization design of the crane box girder is realized based on the improved polyclonal selection algorithm.*
Keywords: *Polyclonal selection, Negative selection, Auto-tolerance, Crane, Girder*

I. Introduction

With the development of the industrial level, in the process of designing bridge crane products, the size of girder usually is chosen according to the crane design manual; however the safety factor of the crane is too large, which causes the waste of the material and increase of the capital costs [1]. As the crane is developing towards the large tonnage, low noise, small vibration and light weight, more and more original products are not suitable for the current requirements [2]. Therefore, Bi et al. [3] firstly finished the global optimization design of the box girders by using the genetic algorithm (GA). Guo et al. [4] also achieved better optimization results based on the improved chaos genetic algorithm. However, the optimization ability of the genetic algorithm is not very strong. In order to improve the search performance of GA, Luo et al. [5] presented an improved clonal selection algorithm by incorporating chaos optimization, and also achieved good results; however, its solution efficiency is reduced. In order to further improve the immune optimization efficiency during the structure design of the box girder in this paper, an improved polyclonal selection algorithm based on negative selection (PSABNS) is put forward, and the test results also show the effectiveness of the proposed algorithm.

II. Polyclonal Selection Based on the Negative Selection

Negative selection is the selection of the immature immune cells. The cells that are able to react to the body's own tissue are removed and other mature cells play an immune role.

Clonal selection theory was presented firstly in 1958 by Burnet according to the cell clone in the animal bodies. Jiao et al. [6] put forward the monoclonal operator and polyclonal operator respectively according to the clonal selection model proposed by L.N.De Castro. The simulation results showed that the convergence of the polyclonal selection algorithm is higher; however its diversity is poor. In order to solve this problem, Du et al. [7] proposed an immune memory clonal planning algorithm that runs independently with antibody population based on the memory unit. While the high convergence speed is kept, the diversity of the population is increased. In this paper, in order to further improve the diversity and the searching capabilities of the polyclonal selection algorithm, the negative selection algorithm [8] is introduced into the polyclonal selection algorithm.

2.1 Polyclonal selection algorithm

In the polyclonal selection algorithm, the crossover operator ($\mathbf{C_r}$) is added on the basis of the operation process, namely clone ($\mathbf{C_l}$), mutation ($\mathbf{M_u}$), selection ($\mathbf{S_e}$), of the monoclonal selection algorithm. The child individual no longer singly inherits the characteristics of the parent individual, but obtains more genetic information, such as the large affinity, etc. from the excellent parents who participate in the crossover operation. The operation is similar to the Lamarck effect and improves the convergence speed [6]. Fig.1 shows the operation process of the polyclonal selection algorithm.

$\forall s_i(k) \in \mathbf{S}(k)$, the main role of $\mathbf{C_l}$ operator is to clone the population antibodies $s_i(k)$ under the clone scale of $\mathbf{q_i}$, and form the population antibodies $s_i'(k)$, that is to say,

$$\boldsymbol{S}'(k) \leftarrow \boldsymbol{C_l}(\boldsymbol{S}(k)) = \{\boldsymbol{S}_1'(k), \boldsymbol{S}_2'(k), \cdots, \boldsymbol{S}_n'(k)\} \quad (1)$$

*** Corresponding author.** E-mail: sheny456@hotmail.com

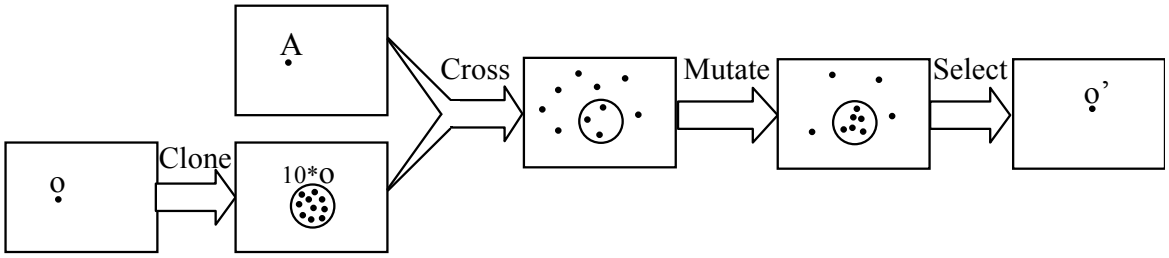


Fig.1 Operation process of the polyclonal selection algorithm

Where,

$\boldsymbol{S}_i'(k) = \{S_{ij}'(k)\} = \{s_{i1}'(k), s_{i2}'(k), \cdots, s_{iq_i}'(k)\}$, $S_{ij}'(k) = s_{ij}'(k) = s_i(k); i = 1,2,\cdots,n; j = 1,2,\cdots,q_i$.

$$q_i(k) = Int[C \cdot \frac{f^{-1}(s_i(k)) \cdot \Theta_i}{\sum_{j=1}^{n} f^{-1}(s_j(k))}] \tag{2}$$

The main role of $\mathbf{C_r}$ operator is to have a crossover on the population $\boldsymbol{S}'(k)$. In order to retain the information of original antibodies, the cross is not acted on the population $\boldsymbol{S}(k)$, namely,

$$\boldsymbol{R}(k) \leftarrow \boldsymbol{C}_r(\boldsymbol{S}'(k)) = \{\boldsymbol{R}_1(k), \boldsymbol{R}_2(k), \cdots, \boldsymbol{R}_n(k)\} \tag{3}$$

$$\boldsymbol{R}_i(k) = \{R_{ij}(k)\} = \{r_{i1}(k), r_{i2}(k), \cdots, r_{iq_i}(k)\} \tag{4}$$

Where, $r_{ij} = s_{ij}'^{1:p}(k) \bigcap s_t^{q:B_I}(k)$; $i \neq t$; $i = 1,2,\cdots,n$; $j = 1,2,\cdots,q_i$.

$\forall r_{ij}(k) \in \boldsymbol{R}(k)$, the main effect of $\mathbf{M_u}$ operator is to let the population $\boldsymbol{R}(k)$ mutate according to the adaptive mutation probability [12], namely,

$$\boldsymbol{R}'(k) \leftarrow \boldsymbol{M}_u(\boldsymbol{R}(k)) = \overline{r_{ij}(k)^{rand(1,sum(B_I)) < p_m^i}} \mapsto r_{ij}(k)^{rand(1,sum(B_I)) < p_m^i} \tag{5}$$

Where, $r_{ij}(k) \in \boldsymbol{R}(k), i \in [1,n], j \in [1,q_i]$.

$\forall\ s_i(k) \in \boldsymbol{S}(k)$ and $\forall\ r_{ij}'(k) \in \boldsymbol{R}'(k)$, the main function of $\mathbf{S_e}$ operator is to select the better individuals among the population $\boldsymbol{R}'(k)$ and the original populations $\boldsymbol{S}(k)$ into the next generation, namely,

$$\boldsymbol{S}(k+1) \leftarrow \boldsymbol{S}_e(\boldsymbol{R}'(k), \boldsymbol{S}(k)) = \{s_1(k+1), s_2(k+1), \cdots, s_n(k+1)\} \tag{6}$$

$$s_i(k+1) = \max\{f^{-1}(r_{ij}'(k)) \mid j \in [1,q_i], f^{-1}(s_i(k))\} \tag{7}$$

The pseudo-code of polyclonal selection algorithm can be described as:

Start:
Initialization: population size n, maximum iterative generation k_{max}, crossover probability p_C, $k \leftarrow 0$, etc.
Generate initial population:

$\boldsymbol{S}(0) = [s_1(0)\ s_2(0) \cdots s_n(0)]$;

While $\mathbf{k \le k_{max}}$

Calculate the antibody affinity $\mathbf{\Theta_i}$;

Calculate the antibody clone size $\mathbf{q_i}$;

$\boldsymbol{S}'(k) \leftarrow \boldsymbol{C}_l(\boldsymbol{S}(k))$;

$\boldsymbol{R}(k) \leftarrow \boldsymbol{C}_r(\boldsymbol{S}'(k))$;

$\boldsymbol{R}'(k) \leftarrow \boldsymbol{M}_u(\boldsymbol{R}(k))$;

$\boldsymbol{S}(k+1) \leftarrow \boldsymbol{S}_e(\boldsymbol{R}'(k), \boldsymbol{S}(k))$;

$k = k+1$;

E nd While

End

2.2 Optimization operators based on the negative selection

2.2.1 The related definitions of the optimization operators

In order to describe the optimization operators conveniently, some related definitions are given firstly.

Definition 1 Under certain matching rules, if the similarity between the strings **a** and **b** exceeds the matching threshold, we call that **a** is match with **b**[8] and denote $\mathbf{M_a}$ (**a**, **b**).

Definition 2 There are two binary strings whose lengths are **l**. If they have the same characters at **r** or more than **r** positions, that is to say, the hamming distance of the two strings is not more than **l-r**, we call that strings **a** is match with **b** under the rules of hamming distance.

Definition 3 Let **r** be the matching threshold. In this paper, the matching threshold is the average value of the antibody population matching rate.

Definition 4 The affinity among a_i and other antibodies is described as:

$$\Theta_i = \min\{\exp(\frac{\| a_i - a_j \|}{B_I} - 1)\}, i \neq j;\ i,j = 1,2,\cdots,\mathrm{n} \tag{8}$$

Where, $\| \bullet \|$ is the hamming distance, and B_I is the number of the coding digits. Obviously, the bigger the antibody affinity, the smaller the Θ_i [6].

2.2.2 Clone deletion and clone supply operators

(1) Clone deletion operator

Step 1 Execute the **T** operation to sort the initial population **S**(k) according to the affinity and achieve the new population.

$$\boldsymbol{S}'(k) \leftarrow \boldsymbol{T}(\boldsymbol{S}(k)) = \boldsymbol{T}(\{s_1(k), s_2(k), \cdots, s_n(k)\}) = \{s_1'(k), s_2'(k), \cdots, s_n'(k)\} \tag{9}$$

$$\forall 1 \le i < j \le n,\ f^{-1}(s_i'(k)) \ge f^{-1}(s_j'(k)).$$

Step 2 Calculate the matching degree among different antibodies. The matching degree of a single antibody can be defined as:

$$\boldsymbol{M_a}(s_i'(k)) = 1 - \Theta_{s_i'(k)} \tag{10}$$

Step 3 Clone deletion. Let a be the matching threshold. Execute the clone deletion operation **D** towards the individuals whose matching degree is higher than the threshold in the later 50% of the sorted individuals.

$$\begin{aligned}\boldsymbol{S}''(k) \leftarrow \boldsymbol{D}(\boldsymbol{S}'(k)) &= \boldsymbol{D}(\{s_1'(k), s_2'(k)\cdots s_{n'}'(k)\cdots s_n'(k)\}) \\ &= \{s_1'(k), s_2'(k)\cdots s_{n'}'(k), s_{n'+1}''(k)\cdots s_m''(k)\}\end{aligned} \tag{11}$$

(2) Clone supply operator

Step 1 Randomly generate a new population $\boldsymbol{N}_e(k)$ with a certain scale, and calculate the matching degree between each antibody and the current optimal solution.

Step 2 Execute the sorting operation **T** towards the new population according to the matching degree to get a new population.

$$\boldsymbol{N}_e'(k) = \boldsymbol{T}(\boldsymbol{N}_e(k)) = \boldsymbol{T}(\{n_e^1(k), n_e^2(k), \cdots, n_e^n(k)\}) = \{n_e'^1(k), n_e'^2(k), \cdots, n_e'^n(k)\} \tag{12}$$

$$\forall 1 \le i < j \le n,\ \boldsymbol{M}_a(n_e'^i(k)) \ge \boldsymbol{M}_a(n_e'^j(k)).$$

Step 3 Clone supply. Pick out the antibodies which are lacked due to the clone deletion from the sorted populations, then the new initial population is got after clone supply.

$$\boldsymbol{S}(k+1) = \boldsymbol{S}''(k) \bigcup \boldsymbol{S}^o(k) = \{s_1'(k), s_2'(k)\cdots s_{n'}'(k), s_{n'+1}''(k), s_m''(k), s_{m+1}^o(k)\cdots s_n^o(k)\} \tag{13}$$

Where, antibody $s^o(k)$ is selected from $\boldsymbol{N}_e'(k)$.

2.3 Flow of the negative selection-based polyclonal selection algorithm

(1) Initializing the population size n, maximum iteration generation k_{max}, crossover probability p_C, variable precision o_p, function variable range b_o, clone scale C and so on. $k \leftarrow 0$.

(2) Initializing the population.

$$\boldsymbol{A}(0) = \{a_1(0), a_2(0), \cdots, a_n(0)\} \in \Psi^{B_I},\ \Psi = \{0,1\} \tag{14}$$

(3) Execute the clone deletion operation towards $\boldsymbol{A}(k)$: ①Calculate the antibody affinity; ②Sort the initial population **A**(k) according to the affinity, namely: $\boldsymbol{A}'(k) \leftarrow \boldsymbol{T}(\boldsymbol{A}(k))$; ③Calculate the antibody matching degree; ④Execute the clone deletion: $\boldsymbol{S}(k) \leftarrow \boldsymbol{D}(\boldsymbol{A}'(k))$.

(4) Execute the clone operation: $\boldsymbol{S}'(k) \leftarrow \boldsymbol{C}_l(\boldsymbol{S}(k))$;

(5) Execute the Crossover operation: $\boldsymbol{R}(k) \leftarrow \boldsymbol{C}_r(\boldsymbol{S}'(k))$;

(6) Execute the mutation operation: $\boldsymbol{R}'(k) \leftarrow \boldsymbol{M}_u(\boldsymbol{R}(k))$;

(7) Execute the selecting operation: $\boldsymbol{S}''(k) \leftarrow \boldsymbol{S}_e(\boldsymbol{R}'(k), \boldsymbol{A}'(k))$.

(8) Execute the clone supply towards $\boldsymbol{S}''(k)$. ①Randomly generate a certain scale of new population $\boldsymbol{N}_e(k)$; ②Calculate the matching degree $\boldsymbol{M}_a(b_e(k), \boldsymbol{N}_e(k))$ between the each antibody of new population and the current optimal Solution $b_e(k)$; ③Sort the new population according to the matching degree; ④Execute the clone supply: $\boldsymbol{S}(k+1) = \boldsymbol{S}''(k) \bigcup \boldsymbol{S}^o(k)$.

(9) Judging whether the termination condition $k = k_{max}$ is met. If not, go on the following operation, otherwise end and output the optimal solution.

(10) $k \leftarrow k+1$ and return to (3).

III. Function Optimization Test and Analysis

In order to verify the optimization performance of the polyclonal selection algorithm based on negative selection (PSABNS), the following six functions are provided to test on a computer by using Matlab7.11.0. The test results are compared with those of the GA and PSA.

1. Ackley's Path function

$$\min f_1 = -20\exp(-0.2sqrt(0.5(x_1^2 + x_2^2))) - \exp(0.5(\cos(2\pi x_1) + \cos(2\pi x_2))) + \exp(1) + 20 \quad (15)$$

$x_i \in$ [-5.12,5.12], $1 \le i \le 2$, $f^* = 0$.

2. Rastrigrin function

$$\min f_2 = 20 + x_1^2 - 10\cos(2\pi x_1) + x_2^2 - 10\cos(2\pi x_2) \quad (16)$$

$x_i \in$ [-5.12, 5.12], $1 \le i \le 2$, $f^* = 0$.

3. Rosenbrock function

$$\min f_3 = 100(x_2 - x_1^2)^2 + (1 - x_1)^2 \quad (17)$$

$x_i \in$ [-5.12, 5.12], $1 \le i \le 2$, $f^* = 0$.

4. Goldstein-Price's function

$$\min f_4 = (1 + (x_1 + x_2 + 1)^2 (19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3(x_2^2)))$$
$$(30 + ((2x_1 - 3x_2)^2)(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)) \quad (18)$$

$x_i \in$ [-2, 2], $1 \le i \le 2$, $f^* = 3$.

5. The multi-peak Function

$$\min f_5 = -(1 + x\sin(4\pi x_1) - y\sin(4\pi x_2 + \pi) + \sin(6sqrt(x_1^2 + x_2^2))/(6sqrt(x_1^2 + x_2^2 + 1\times 10^{-15}))) \quad (19)$$

$x_i \in$ [-1, 1], $1 \le i \le 2$, $f^* = -2.118$.

6. Schaffer's Function

$$\min f_6 = 0.5 + ((\sin(sqrt(x_1^2 + x_2^2)))^2 - 0.5)/((1 + 0.0001(x_1^2 + x_2^2)^2)^2) \quad (20)$$

$x_i \in$ [-10, 10], $1 \le i \le 2$, $f^* = 0$.

Considering the randomness of the intelligent algorithm, each function is tested 50 times independently.

Table 1 Comparison results among three algorithms

f	ε	N_{best}			N_{max}			N_{mean}		
		GA	PSA	PSABNS	GA	PSA	PSABNS	GA	PSA	PSABNS
f_1	1.0E-06	50	50	50	805	299	210	403.12	112.04	80
f_2	1.0E-02	50	50	50	184	141	37	62.56	30	14.62
f_3	1.0E-03	14	32	49	950	498	336	347.4286	76.5938	84.0204
f_4	1.0E-06	37	50	50	993	89	61	474.8649	38.26	30.38
f_5	1.0E-03	13	50	50	855	86	28	421.7692	13.42	9.58
f_6	1.0E-03	22	50	48	960	823	508	335.4545	74.54	73.2083

Table 1 is the comparison results among three algorithms. N_{best} denotes the number of times an optimal solution was found. N_{max} and N_{mean} denote the maximum and average convergence generations needed to find the optimal solutions, respectively. From the three performance index in table 1, it can be seen that the optimization results of PSABNS are better than those of GA and PSA, which shows the strong optimization ability of the proposed negative selection-based polyclonal algorithm.

IV. Optimization Design of Crane Box Girder

4.1 Construction of optimization model

In order to further verify the validity of the negative selection-based polyclonal algorithm, the proposed algorithm is used in the optimization design of the crane box girder. Fig. 2 shows the cross section of the crane box girder. The minimal mass of the girder is taken as the optimization objective and the optimization model is constructed as follows:

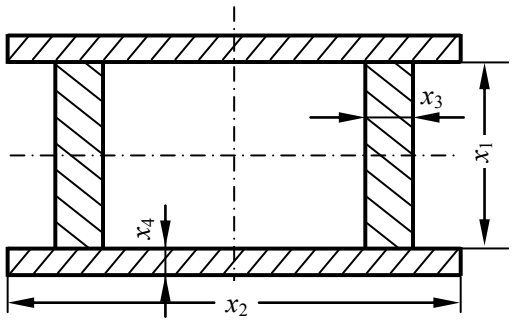


Fig.2 Cross section of crane box girder

(1) Designing variables

$$x = \{x_1, x_2, x_3, x_4\} \tag{21}$$

Where, x_1 is the height of the main girder; x_2 is the width of the main girder; x_3 is the thickness of the web; x_4 is the thickness of the wing plate.

(2) Objective function

$$f(x) = 2(x_1x_3 + x_2x_4) \tag{22}$$

(3) Constraints including strength, stiffness, process and boundary constraints.

$$g_1(x) = \frac{3L}{4}\left[\frac{P_1 + r(x_1x_3 + x_2x_4)L}{3x_1x_2x_4 + x_1^2x_3} + \frac{P_2}{3x_1x_2x_4 + x_2^2x_4}\right] - [\sigma] \le 0 \tag{23}$$

$$g_2(x) = \frac{p_1L^3}{(3x_1^2x_2x_4 + x_1^3x_3)\times 9.9\times 10^6} - \frac{L}{700} \le 0 \tag{24}$$

$$g_3(x) = \frac{x_2}{x_4} - 60 \le 0 \tag{25}$$

$$g_4(x) = \frac{x_1}{x_3} - 160 \le 0 \tag{26}$$

$$g_5(x) = 0.5 - x_3 \le 0 \tag{27}$$

$$g_6(x) = 0.5 - x_4 \le 0 \tag{28}$$

Where, P_1=1.2×10^5, P_2=1.2×10^4, L=10.5m, [σ]=140MPa, 700≤x_1≤800, 350≤x_2≤400, 5≤x_3≤10, 5≤x_4≤10.

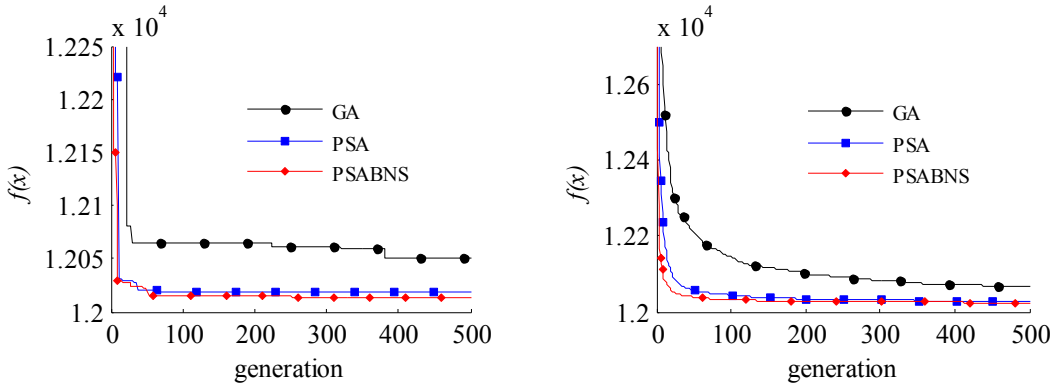
4.2 Optimization results and their analysis

Table 2 gives the optimization comparison among four algorithms, namely, feasible direction method (FDM), GA, PSA and the proposed PSABNS. From the table, we can see that the optimization results of three intelligent algorithms (namely, GA, PSA and PSABNS) are better than the result of FDM for their global optimization ability. The optimization results of PSA and PSABNS are better than the result of GA because the clone operation improves the optimization ability of intelligent algorithms. Moreover, the optimization result of the proposed PSABNS is the best among the four algorithms because the introduction of negative selection further improves the optimization ability of the polyclonal selection algorithm.

Table 2 Optimization comparison among four algorithms

index \ method	FDM	GA		PSA		PSABNS	
		Optimal	Average	Optimal	Average	Optimal	Average
$\min f(x)$	12860	12051	12065	12018	12029	12014	12024
$N_{hg}(x_1)$	790	751.0031	/	778.9556	/	784.719	/
$N_{wg}(x_2)$	310	358.2253	/	351.3737	/	350.0172	/
$N_{tw}(x_3)$	5	5.0001	/	5.0006	/	5.0001	/
$N_{tp}(x_4)$	8	6.3377	/	6.016	/	5.9519	/

Fig. 3 gives the evolutionary curves of three intelligent optimization algorithms. From the figure, it can be seen that whether the evolutionary curves are for optimal solution or for the whole population, the results of the PSABNS are the best among the three optimization algorithms, which further verifies the validity of the proposed negative selection-based polyclonal algorithm.



(a) Optimal solution (b) Whole population

Fig.3 Evolutionary curves of three intelligent algorithms

V. Conclusion

In order to solve the optimization of the bridge crane box girder, on the basis of the polyclonal selection algorithm, an improved polyclonal selection algorithm based on negative selection is proposed in this paper. By taking advantages of the clone deletion and supply operation based on negative selection, the optimization ability of the proposed algorithm is improved. The experiment results of the typical function test and the crane box girder optimization verify the effectiveness of the PSABNS. Compared with other related optimization algorithm, negative selection-based polyclonal selection algorithm has better optimization ability in solving the multi-constraints box girder optimization problem.

Acknowledgements

This work is supported by the Modern Educational Technology Project of Jiangsu Province (No. 2013-R-24771), 2013 Teaching Reform and Teaching Study Project from Jiangsu University of Science and Technology (JUST), 2013 Key Teaching Reform and Teaching Study Project from Zhangjiagang campus areas of JUST and 2013 Key Course building Project from Suzhou Institute of Technology.

References

[1] Y. Shen and Y.F. Bu, PSO optimization of crane box grider based on chaos series, Hoisting and Conveying Machinery, 11, 2009,55-58.
[2] X.F. Cheng, Optimization of main girder section for overhead crane, Journal of East China Jiaotong University, 21(4), 2004,113-115.
[3] C.C. Bi and Y.Z. Ding, Application of genetic algorithm in the optimization design for the crane box grider, Hoisting and Conveying Machinery, 9, 1999,5-7.
[4] H.X. Guo, X.Y. Che and W.Y. Xiao, Chaos-genetic optimal algorithm and application in mechanical optimal design, Journal of machine design, 20(10), 2003, 23-25.
[5] M. Luo, L. Lei and C.C. Wei, Optimization design of hybrid chaos immune algorithm with self-adaptive parameter adjusting, Journal of Xinyang Normal University, 25(3), 2012, 387-390.
[6] L.C. Jiao, H.F. Du and F. Liu, Calculation, learning and recognition of immune optimization, Beijing, Science Press, 2006.
[7] H.F. Du, M.G. Gong and L.C. Jiao, Immune memory clonal programming algorithm for high-dimension,Progress in Natural Science,14(8), 2004, 925-933.
[8] H. Zhang, L.F. Wu and Y.S. Zhang, An algorithm of r-adjustable negative selection algorithm and its simulation analysis, Chinese Journal of Computers, 28(10), 2005, 1614-1619.