The Design Optimization Problem of Welded Beam Design Studies

Zhiqing Chen

Department of electrical and Electronic Engineering, Wenzhou Polytechnic, Wenzhou 325035, China.

Abstract

Many advanced algorithms have been applied to the optimal design of welded beams.In order to improve the convergence speed of the algorithm and balance global exploration and local exploitation, an enhanced algorithm WGBO is proposed, which includes gradient search rule (GSR) and local escape operator (LEO) with adaptive weights.In addition, the improved algorithm is applied to the design optimization of welded beams, and the results show that the WGBO algorithm has a certain improvement significance.

Keywords

Gradient Based Optimizer; Meta-heuristic; Welded Beam Design.

1. Introduction

In the GBO is a new meta-heuristic optimization based on gradient and population. It mainly uses a set of vectors and two operators to explore the entire search space.

The vector is expressed as follows.

$$X_{n,d} = [X_{n,1}, X_{n,2}, \dots, X_{n,D}], \quad n = 1,2,\dots,N \quad d = 1,2,\dots,D \tag{1}$$

The initial population of GBO can be expressed by Eq. (2).

$$X_n = X_{min} + rand(0,1) \times (X_{max} - X_{min}) \tag{2}$$

where X_{min} is the minimum value of the X bound, X_{max} is the maximum value of the X bound, and $rand(0,1)$ is a random number between 0 and 1.

The specific GSR equation is as follows

$$GSR = randn \times \rho_1 \times \frac{2\Delta x \times x_n}{(x_{worst} - x_{best} + \varepsilon)} \tag{3}$$

$$\rho_1 = 2 \times rand \times \alpha - \alpha \tag{4}$$

$$\alpha = |\beta \times \sin(\frac{3\pi}{2} + \sin(\beta \times \frac{3\pi}{2}))| \tag{5}$$

$$\beta = \beta_{min} + (\beta_{max} - \beta_{min}) \times (1 - (\frac{m}{M})) \tag{6}$$

where β_{min} and β_{max} are equal to 0.2 and 1.2 respectively, m is the value of current evaluations, M is the total evaluations, α is a function determined by β, ε is a random number between 0 and 0.1.

The expression of Δx is defined as follow.

$$\Delta x = rand(1:N) \times |step| \tag{7}$$

$$step = \frac{(x_{best} - x_{r1}^m) + \delta}{2} \tag{8}$$

$$\delta = 2 \times rand \times (|\frac{x_{r1}^m + x_{r2}^m + x_{r3}^m + x_{r4}^m}{4} - x_n^m|) \tag{9}$$

where $rand(1:N)$ is a random number with N dimensions, $r1, r2, r3$ and $r4$ are four different random numbers selecting from [1, N], and $step$ is a step size.

DM is defined as follows.

$$DM = rand \times \rho_2 \times (x_{best} - x_n) \tag{10}$$

The definition of ρ_2 is shown as follows.

$$\rho_2 = 2 \times rand \times \alpha - \alpha \tag{11}$$

Finally, the defination of GSR is expressed as follow.

$$x1_n^m = x_n^m - randn \times \rho_1 \times \frac{2\Delta x \times x_n^m}{(x_{worst} - x_{best} + \varepsilon)} + rand \times \rho_2 \times (x_{best} - x_n^m) \tag{12}$$

And the new vector $x2_n^m$ is expressed as follow.

$$x2_n^m = x_{best} - randn \times \rho_1 \frac{2\Delta x \times x_n^m}{(x_{worst} - x_{best} + \varepsilon)} + rand \times \rho_2 \times (x_{r1}^m - x_{r2}^m) \tag{13}$$

the new solution to x_n^{m+1} is expressed as follow.

$$x3_n^m = x_n^m - \rho_1 \times (x2_n^m - x1_n^m) \tag{14}$$

$$x_n^{m+1} = r_a \times (r_b \times x1_n^m + (1 - r_b) \times x2_n^m) + (1 - r_a) \times x3_n^m \tag{15}$$

where r_a and r_b are a random number between 0 and 1.

The solution to X_n^{m+1} is updated by the common effect of the best solution x_{best}, the solution $X1_n^m$ and $X2_n^m$, the interaction of two random solution x_{r1}^m and x_{r2}^m., and a random gained solution x_k^m. The definition of pr decides the execution probability of GSR. The equation of LEO is expressed as follows.

$$X_n^{m+1} = X_n^{m+1} + f_1 \times (u_1 \times x_{best} - u_2 \times x_k^m) + f_2 \times \rho_1 \times (u_3 \times (X2_n^m - X1_n^m) + u_2 \times (x_{r1}^m - x_{r2}^m))/2. \quad rand<0.5 \tag{16}$$

$$X_n^{m+1} = x_{best} + f_1 \times (u_1 \times x_{best} - u_2 \times x_k^m) + f_2 \times \rho_1 \times (u_3 \times (X2_n^m - X1_n^m) + u_2 \times (x_{r1}^m - x_{r2}^m))/2. \quad rand\geq 0.5 \tag{17}$$

where f_1 is a random number between -1 and 1, f_2 is a random from a normal distribution, and u_1, u_2 and u_3 are expressed as follows.

$$u_1 = L_1 \times 2 \times rand + (1 - L_1) \tag{18}$$

$$u_2 = L_1 \times rand + (1 - L_1) \tag{19}$$

$$u_3 = L_1 \times rand + (1 - L_1) \tag{20}$$

In above, *rand* is a random number between 0 and 1, L_1 is a binary parameter with the value of 0 or 1.

The above x_k^m is defined as follow.

$$x_k^m = \begin{cases} x_{rand} & u_2 < 0.5 \\ x_p^m & u_2 \geq 0.5 \end{cases} \tag{21}$$

$$x_{rand} = X_{min} + rand \times (X_{max} - X_{min}) \tag{22}$$

where x_{rand} is a new solution, x_p^m is a random selected solution.

Among them, FES is the current calculation amount and MaxFEs is the max calculation amount respectively. All the experiments in this paper, MaxFEs is 300,000. The variables are changing continuously. When the optimal solution position of the population is not updated, the value is increased by 1 automatically. Once the Xnew can't jump out in the local scope, it is easy to cause the algorithm unable to carry out a larger global search, unable to find a better solution. Adaptive weight mechanism is introduced into the basic GBO algorithm [1].

Many researchers began to introduce adaptive weights into the optimization algorithms to obtain good results. we introduce a crucial weight pr which acts on LEO. If rand is smaller than pr, the LEO can be defined by a equation. If rand is bigger than pr, the LEO will be defined by another expression. So the adaptive weight pr is helpful for the algorithm to expand the global search in the early stage and avoids falling into the local optimum prematurely. Additionally, search the local range carefully in the later stage to improve the accuracy of the solution. The mathematical formula of pr is expressed as follows.

$$pr = FES/MaxIt \tag{23}$$

$$Pr = w \tag{24}$$

FES is the current evaluation and MaxIt is the max evaluation number respectively. All the experiments in this paper, MaxIt is 300,000. While escaping the local optimum, the variables are continuously changing.

Table 1. Comparison of results for different algorithms

Algorithm	WGBO	ESSA	WLSSA	CBA	CDLOBA	CMAES
+/=/-		28/1/1	14/5/11	26/0/4	27/1/2	14/15/1
ARV	1.933333	4.933333	2.5	4.466667	4.2	2.8

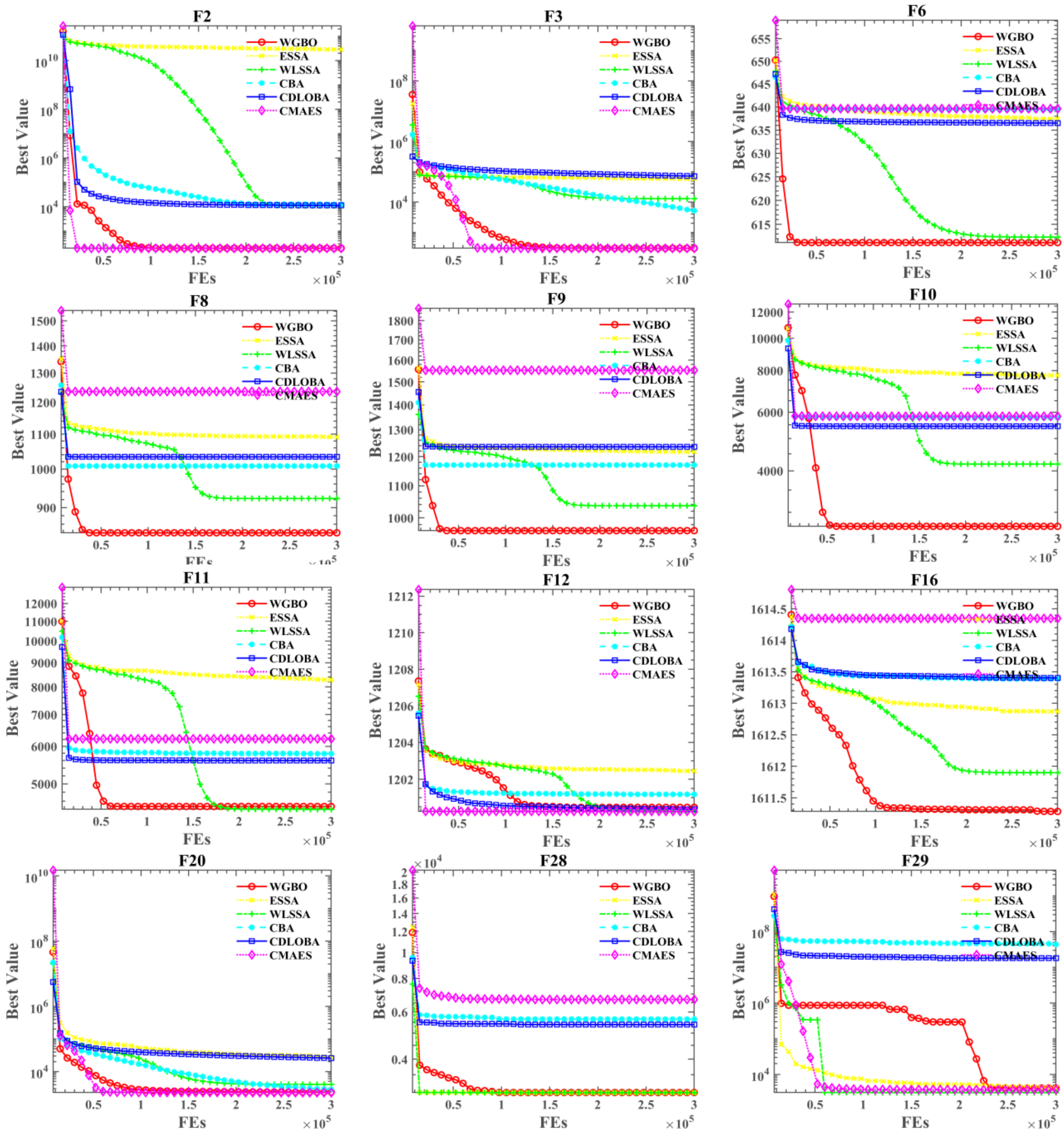

Fig. 1. Convergence trends of WGBO and advanced algorithms.

2. Comparison of WGBO with advanced algorithms analysis

The In this section, the effectiveness of the proposed WGBO is concluded by comparing with the existing advanced algorithms methods on IEEE CEC 2014 test functions. There are five well-known meta-heuristic algorithms compared in the experiment: WLSSA [2], ESSA [3], CBA [4], CDLOBA [5], CMAES [6]. We set the total size of the search space distribution to be 30, the dimension of the solution space to be 30, and the maximum evaluation times to be 300,000. What's more, each algorithm runs independently 30 times on each test function. Table 1 shows the results of ranking of Friedman test on the average performance of the algorithm. It can be seen that the minimum average ranking value (ARV) obtained by the poposed WGBO algorithm in this paper is 1.933333, ranking first among all the algorithms. So we conclude that the proposed WGBO algorithm is quite competitive compared with the five advanced methods on the CEC2014 test functions.

Fig. 1 shows the convergent trend images of these comparison algorithms selected from CEC2014 test functions. Better solution quality is obtained in F6, F8, F9, F10, F16 of WGBO better than the other algorithms. In F2, F3 and F20, the convergence accuracy of WGBO is higher than the other algorithms, and only the CMAES algorithm can reach the same accuracy of WGBO. Among the other test functions, under the influence of the adaptive weight mechanism, WGBO keeps moving towards the global optimal solution and obtains satisfactory results in the later stages.

By introducing adaptive weight and acting simultaneously on LEO stage in the original algorithm, the motion throughout the iterative process is accelerated. Therefore, it expands the scope of global search. The algorithm is enhanced in terms of the accuracy of the solutions and the quickness of convergence significantly. Nor does the proposed version remove any of the benefits of basic GBO in terms of component efficacy. Through the above analysis, we conclude the effectiveness and competitiveness of the proposed WGBO algorithm compared with the other advanced algorithms.

3. Welded beam design

The objective of the WBD problem [7] is to obtain the minimum manufacturing cost of WBD. In this case, it takes shear stress (τ), bending stress (θ), buckling load (P_c), deflection (δ) as the constraints. There are four variants and seven constraints for WBD problem: x1 represents the thickness (h); x2 is the length of the clamped bar (l); x3 represents the height t (t); x4 represents the thickness (b). The optimization model of WBD case is described as follows:

Consider

$$\vec{x} = [x_1\ x_2\ x_3\ x_4] = [h\ l\ t\ b]$$

Minimize

$$f(\vec{x}) = 1.10471x_1^2 + 0.04811x_3x_4(14.0 + x_4)$$

Subject to

$$g_1(\vec{x}) = \tau(\vec{x}) - \tau_{max} \le 0$$

$$g_2(\vec{x}) = \sigma(\vec{x}) - \sigma_{max} \le 0$$

$$g_3(\vec{x}) = \delta(\vec{x}) - \delta_{max} \le 0$$

$$g_4(\vec{x}) = x_1 - x_4 \le 0$$

$$g_5(\vec{x}) = P - P_C(\vec{x}) \le 0$$

$$g_6(\vec{x}) = 0.125 - x_1 \le 0$$

$$g_7(\vec{x}) = 1.10471x_1^2 + 0.04811x_3x_4(14.0 + x_2) - 5.0 \le 0$$

Variable range:

$$0.1 \le x_1 \le 2,\ 0.1 \le x_2 \le 10,\ 0.1 \le x_3 \le 10,\ 0.1 \le x_4 \le 2$$

Where

$$\tau(\vec{x}) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2}\ \ \tau' = \frac{P}{\sqrt{2}x_1x_2}\ \ \tau'' = \frac{MR}{J}\ \ \mathrm{M} = \mathrm{P}(\mathrm{L} + \frac{x_2}{2})$$

$$R = \sqrt{\frac{x_2^2}{4} + (\frac{x_1+x_3}{2})^2}$$

$$J = 2\left\{\sqrt{2}x_1x_2\left[\frac{x_2^2}{4} + (\frac{x_1+x_3}{2})^2\right]\right\}$$

$$\sigma(\vec{x}) = \frac{6PL}{x_4{x_3}^2},\ \ \delta(\vec{x}) = \frac{6PL^3}{Ex_3^2x_4}$$

$$P_C(\vec{x}) = \frac{4.013E\sqrt{\frac{x_3^2x_4^6}{36}}}{L^2}(1 - \frac{x_3}{2L}\sqrt{\frac{E}{4G}})$$

$$P = 60001\mathrm{b},\mathrm{L} = 14\mathrm{in}..\delta_{max} = 0.25\ in..$$

$$E = 30 \times 1^6 psi,\mathrm{G} = 12 \times 10^6 psi$$

$$\tau_{max} = 13600psi,\sigma_{max} = 30000psi$$

The results of WGBO are compared with the other solutions which worked previously in the Table 2. We can see that WGBO achieves the optimal cost of 1.7199. This conclusion expresses that when the four parameters are set to 0.1994033, 3.46006, 9.010189, and 0.207168 respectively, the manufacturing cost of WB can reach 1.7199. WGBO is superior to all other methods. In conclusion, comparing with other algorithms, the proposed WGBO solves this problem effectively and gets the minimum manufacturing cost.

Table 2. Comparison results of WGBO for the WBD problem

Technique	Best variables				Best cost
	h	*l*	*t*	*b*	
WGBO	0.1994033	3.46006	9.010189	0.207168	1.7199
RO[53]	0.203687	3.528467	9.004233	0.207241	1.735344
HS [54]	0.2442	6.2231	8.2915	0.2433	2.3807
IHS [46]	0.20573	3.47049	9.03662	0.20573	1.7248
Random [55]	0.4575	4.7313	5.0853	0.6600	4.1185
Simple [55]	0.2792	5.6256	7.7512	0.2796	2.5307
David [55]	0.2434	6.2552	8.2915	0.2444	2.3841

4. Discussion and Conclusions

This paper proposes an improved GBO based on the adaptive weight which named WGBO. The numerical results of 30 IEEE CEC2014 test functions have shown that the developed enhanced WGBO is superior to five advanced algorithms and obtained the better performance comparing the other counterparts significantly facing welded beam design problems.

References

[1] Ahmadianfar, I., O. Bozorg-Haddad, and X. Chu, Gradient-based optimizer: A new metaheuristic optimization algorithm. Information Sciences, 2020. 540: p. 131-159.

[2] Hao Ren, Jun Li∗, Huiling Chen, ChenYang Li. Adaptive levy-assisted salp swarm algorithm: Analysis and optimization case studies. Mathematics and computers in Simulation 181(2021) 380-409

[3] M. Qais, H. Hasanien, S. Alghuwainem, Enhanced salp swarm algorithm: Application to variable speed wind generators, Eng. Appl. Artif. Intell. 80 (2019) 82–96

[4] B.R. Adarsh, et al., Economic dispatch using chaotic bat algorithm, Energy 96 (2016) 666–675.

[5] J. Yong, et al., A novel bat algorithm based on collaborative and dynamic learning of opposite population, 2018, pp. 541–546. 70.

[6] N. Hansen, A. Ostermeier, Completely derandomized self-adaptation in evolution strategies, Evol. Comput. 11 (2003) 159–195.

[7] Coello, C. A. C. (2000). Use of a self-adaptive penalty approach for engineering optimization problems. Computers in Industry, 41, 113-127.