Check for updates

A greedy non-hierarchical grey wolf optimizer for real-world optimization

Ebrahim Akbari,[1] Abolfazl Rahimnejad,[2] and Stephen Andrew Gadsden[2,✉]

[1]Department of Electrical Engineering, University of Isfahan, Isfahan, Iran

[2]College of Engineering and Physical Sciences, University of Guelph, Guelph, N1G 2W1, Canada

✉Email: gadsden@uoguelph.ca

Grey wolf optimization (GWO) algorithm is a new emerging algorithm that is based on the social hierarchy of grey wolves as well as their hunting and cooperation strategies. Introduced in 2014, this algorithm has been used by a large number of researchers and designers, such that the number of citations to the original paper exceeded many other algorithms. In a recent study by Niu et al., one of the main drawbacks of this algorithm for optimizing real-world problems was introduced. In summary, they showed that GWO's performance degrades as the optimal solution of the problem diverges from 0. In this paper, by introducing a straightforward modification to the original GWO algorithm, that is, neglecting its social hierarchy, the authors were able to largely eliminate this defect and open a new perspective for future use of this algorithm. The efficiency of the proposed method was validated by applying it to benchmark and real-world engineering problems.

Introduction: The Grey wolf optimization (GWO) algorithm, inspired by the hunting behaviour of the grey wolves in the wild, was developed by Mirjalili et al. in 2014 [1]. This algorithm has been used by many researchers in recent years for various optimization problems. For example, Venkatakrishnan et al. have used GWO algorithm to optimally dispatch real power in power systems [2]. Sulaiman et al. have successfully applied this algorithm to the same basic form of reactive power dispatch for the IEEE standard electrical networks [3]. A multi-objective version of GWO algorithm has been used in [4] for attribute reduction. A modified version of GWO was proposed in [5] to enhance wind speed forecasting, and the authors of [6] claimed that they successfully applied this algorithm to an optimization problem for multi-level thresholding. An augmented GWO was proposed in [7] for the optimal design of PI controllers of a grid-connected PMSG, operated by wind turbine and a hybrid cuckoo search and GWO algorithm, was used in [8] for performance enhancement of HVDC-based offshore wind farms by optimal controller design.

In the original GWO algorithm, the three best members ranked based on their objective function values play the roles of α, β, and δ wolves, and the other members are considered the ω wolves. In the original GWO, similar to the social hierarchy of the grey wolves in nature, α, β and δ guide the hunting process, and the positions of the ω wolves are determined by the positions of these three wolves. In this algorithm, the positions of the grey wolves are updated in each iteration using the following equations [1]:

$$\vec{D}_\alpha = \left|\vec{C}_1.\vec{X}_\alpha - \vec{X}_i\right| \tag{1}$$

$$\vec{D}_\beta = \left|\vec{C}_2.\vec{X}_\beta - \vec{X}_i\right| \tag{2}$$

$$\vec{D}_\delta = \left|\vec{C}_3.\vec{X}_\delta - \vec{X}_i\right| \tag{3}$$

$$\vec{X}_1 = \vec{X}_\alpha - \vec{A}_1.(\vec{D}_\alpha) \tag{4}$$

$$\vec{X}_2 = \vec{X}_\beta - \vec{A}_2.(\vec{D}_\beta) \tag{5}$$

$$\vec{X}_3 = \vec{X}_\beta - \vec{A}_3.(\vec{D}_\beta) \tag{6}$$

$$\vec{X}_i(t+1) = (\vec{X}_1 + \vec{X}_2 + \vec{X}_3)/3 \tag{7}$$

In the above equations, t represents the current iteration of the algorithm; $\vec{A}$ and $\vec{C}$ denote the coefficient vectors; $\vec{X}_\alpha$, $\vec{X}_\beta$, $\vec{X}_\delta$ denote the position vectors of α, β, and δ, and $\vec{X}_i$ denotes the position vector of the ith grey wolf. $\vec{A}$ and $\vec{C}$ vectors are calculated as follows:

$$\vec{A} = 2\vec{a}.\vec{R}_1 - \vec{a} \tag{8}$$

$$\vec{C} = 2.\vec{R}_2 \tag{9}$$

where $\vec{a}$ decreases linearly from 2 to 0 during iterations, and $\vec{R}_1$ and $\vec{R}_2$ are vectors of uniformly distributed random numbers in the range of 0 and 1.

Deficiency and defect of GWO: Niu et al. [9] have demonstrated well that the GWO algorithm has good performance compared to most of the other algorithms for basic functions, whose optimal solution is zero; on the other hand, it has poor performance for the shifted functions, whose optimal solution is far from zero. However, most real-world problems have various non-zero optimal points. Therefore, the practical use of this algorithm is strangely restricted. Consequently, it seems that we need a fundamental modification in GWO algorithm to be efficiently applicable to a wide range of real-world problems.

Greedy non-hierarchical GWO (G-NHGWO): In the original GWO algorithm, three of the best solutions are always stored as α, β and δ wolves, and these three members of the population always guide the rest of the population in their update equation, which is analogous to the social hierarchy of the grey wolf packs in nature. Therefore, this algorithm has a good speed in converging to the optimal solution of basic functions with optimal solution of zero. This update mechanism has two major drawbacks in optimizing real-world functions: first, due to the use of the best global solutions found so far, the algorithm converges very quickly to a local optimal solution and loses its optimization power significantly; second, it causes the loss of a variety of new population in each iteration of the algorithm.

In order to fix these two shortcomings and strengthen the GWO algorithm, we have defined and saved the best solution found so far by each grey wolf as its personal best position, like the PSO algorithm [10]; for instance, the personal best position for the ith wolf would be $\vec{X}_i^{best}$. Then, instead of selecting α, β and δ wolves to guide the population updating, i.e. by neglecting the social hierarchy of grey wolf pack, three members r_1, r_2 and r_3 are randomly selected and their positions, i.e. $\vec{X}_{r1}$, $\vec{X}_{r2}$, and $\vec{X}_{r3}$, are used to guide the population update mechanism. For example, the updating equations of the ith member would be as follows:

$$\vec{D}_{r1} = \left|\vec{C}_1.\vec{X}_{r1} - \vec{X}_i\right| \tag{10}$$

$$\vec{D}_{r2} = \left|\vec{C}_2.\vec{X}_{r2} - \vec{X}_i\right| \tag{11}$$

$$\vec{D}_{r3} = \left|\vec{C}_3.\vec{X}_{r3} - \vec{X}_i\right| \tag{12}$$

$$\vec{X}_1 = \vec{X}_{r1} - \vec{A}_1.(\vec{D}_{r1}) \tag{13}$$

$$\vec{X}_2 = \vec{X}_{r2} - \vec{A}_2.(\vec{D}_{r2}) \tag{14}$$

$$\vec{X}_3 = \vec{X}_{r3} - \vec{A}_3.(\vec{D}_{r3}) \tag{15}$$

$$\vec{X}_i'(t+1) = (\vec{X}_1 + \vec{X}_2 + \vec{X}_3)/3 \tag{16}$$

$$\vec{X}_i(t+1) = \begin{cases} \vec{X}_i'(t+1), & \text{if} f(\vec{X}_i'(t+1)) < f(\vec{X}_i(t)) \\ \vec{X}_i(t), & \text{otherwise} \end{cases} \tag{17}$$

It should be noted that in [11] a random walk is proposed for updating the positions of α, β, and δ. However, this is different from our case, in which we use three randomly selected members from the population and use their positions as the new leaders. Furthermore, the proposed algorithm is a greedy-based method, and hence, the grey wolves move to a new position only if it is better than their current position. In other words, in the proposed G-NHGWO method all grey wolves are always located at the best positions they have found so far.

Table 1. The optimal results of GWO and the proposed G-NHGWO algorithms on the 30-D real-parameter test functions

F	GWO Mean Std Dev	G-NHGWO Mean Std Dev	%Imp Winner
F1	2.03E+03 2.67E+03	6.45E-01 6.71E-01	99.97 +
F2	1.34E+04 5.95E+03	7.50E+02 5.44E+02	94.40 +
F3	1.78E+07 7.86E+06	4.59E+06 1.76E+06	74.21 +
F4	1.41E+04 2.14E+03	1.11E+03 7.33E+02	92.13 +
F5	5.44E+03 3.65E+03	8.25E+02 1.29E+02	84.83 +
F6	1.06E+08 2.26E+08	8.74E+03 1.79E+04	99.99 +
F7	9.84E+01 8.44E+01	1.92E+00 4.75E-01	98.05 +
F8	2.10E+01 4.06E-02	2.09E+01 3.82E-02	0.48 +
F9	1.05E+02 1.48E+01	1.32E+01 3.09E+00	87.43 +
F10	2.07E+02 6.74E+01	6.99E+01 7.10E+01	66.23 +
F11	1.87E+01 2.53E+00	2.82E+01 1.19E+01	−50.80 -
F12	9.29E+04 3.54E+04	1.68E+04 1.63E+04	81.92 +
F13	3.83E+00 7.07E-01	2.93E+00 4.56E-01	23.50 +
F14	1.16E+01 4.28E-01	1.09E+01 3.03E-01	6.03 +

Table 2. The statistical results of GWO and the proposed G-NHGWO algorithms for solving the ELD problem in different test power systems

Test system	GWO Mean Best Std Dev	G-NHGWO Mean Best Std Dev	%Imp Winner
6-gen. system [14]	608.67 606.03 5.23	606.25 606.03 0.16	0.398 +
20-gen. system [15]	62,525.08 62,468.96 45.96	62,463.52 62,456.36 4.72	0.098 +
40-gen. system [16]	122,684.42 122,152.01 434.02	122,143.81 121,908.41 153.23	0.441 +

Results and discussion: In the first part of the simulation studies, we have selected 14 test functions from cec2005 [12] to demonstrate the power and effectiveness of the proposed modified algorithm compared to the original algorithm. Functions 1 to 14, which represent the real-world problems having shifted functions, have been successfully implemented in many articles [13]. Functions 1 to 6 are unimodal functions, functions 7 to 12 are multimodal functions, and functions 13 and 14 are expanded multimodal functions.

The number of population for both algorithms has been set to 30, based on the original reference [1], and the number of iterations has also been set to 10,000. Hence, the number of function evaluations for both algorithms is equal to 3,00,000, which is exactly equal to the value proposed by the cec2005 [12]. A total of 25 independent runs were carried out for optimizing each test function by each algorithm, and the result over all runs, including the mean and standard deviation, are presented in Table 1.GWO (−). In this table, %Imp shows the percentage of improvement of the mean index through using the G-NHGWO with respect to

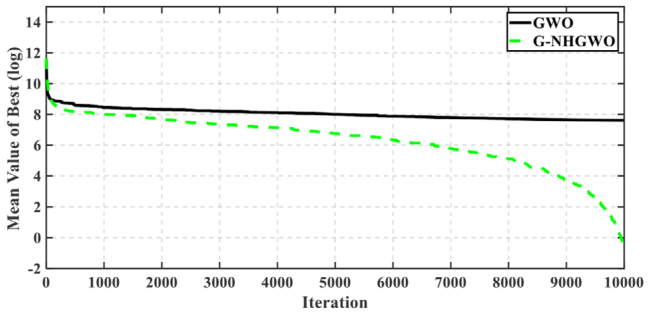
Table 3. The best solution found by GWO and G-NHGWO algorithms for all test systems

Parameter (MW)	6-gen. system [14]		20-gen. system [15]		40-gen. system [16]	
	GWO	G-NHGWO	GWO	G-NHGWO	GWO	G-NHGWO
P_{g1}	11.86	12.34	518.11	509.33	113.72	111.97
P_{g2}	28.2	27.23	167.58	157.18	113.44	110.74
P_{g3}	58.83	58.11	135.6	131.45	99.98	98.83
P_{g4}	98.5	98.61	86.45	102.41	188.19	179.73
P_{g5}	52.01	52.88	100.98	106.61	93.17	89.78
P_{g6}	36.55	36.78	47.42	62.87	139.86	136.79
P_{g7}	-	-	117.05	102.14	278.29	259.85
P_{g8}	-	-	103.17	121.43	288.33	285.21
P_{g9}	-	-	112.95	112.48	287.05	286.47
P_{g10}	-	-	109.28	116.1	131.04	134.41
P_{g11}	-	-	155.22	151.85	124.66	168.4
P_{g12}	-	-	300.19	303.2	170.3	164.5
P_{g13}	-	-	126.37	122.53	214.71	214.93
P_{g14}	-	-	34.9	51.12	305.35	305.04
P_{g15}	-	-	122.81	116.16	394.45	394.23
P_{g16}	-	-	39.72	36.13	304.84	394.53
P_{g17}	-	-	65.5	56.98	489.96	489.45
P_{g18}	-	-	102.71	90.23	489.66	490.05
P_{g19}	-	-	116.93	95.18	511.56	512.38
P_{g20}	-	-	32.02	47.09	511.99	512.63
P_{g21}	-	-	-	-	524.2	524.48
P_{g22}	-	-	-	-	523.34	523.3
P_{g23}	-	-	-	-	527.12	523.56
P_{g24}	-	-	-	-	523.81	523.85
P_{g25}	-	-	-	-	532.69	523.6
P_{g26}	-	-	-	-	523.85	524.08
P_{g27}	-	-	-	-	11.22	11.14
P_{g28}	-	-	-	-	11.33	10.69
P_{g29}	-	-	-	-	11.51	13
P_{g30}	-	-	-	-	93.5	89.47
P_{g31}	-	-	-	-	168.55	188.33
P_{g32}	-	-	-	-	189.91	188.21
P_{g33}	-	-	-	-	189.9	189.77
P_{g34}	-	-	-	-	199.84	167.65
P_{g35}	-	-	-	-	179.13	182.88
P_{g36}	-	-	-	-	199.23	179.5
P_{g37}	-	-	-	-	109.95	98.58
P_{g38}	-	-	-	-	107.92	93.67
P_{g39}	-	-	-	-	109.79	92.28
P_{g40}	-	-	-	-	512.64	512.06
Total generation	285.94	285.95	2594.96	2592.47	10500	10500
Load	283.4	283.4	2500	2500	10500	10500
Power loss	2.54	2.55	94.96	92.46	-	-
Power mismatch	0.0010	0	0.0023	0.0124	0	0

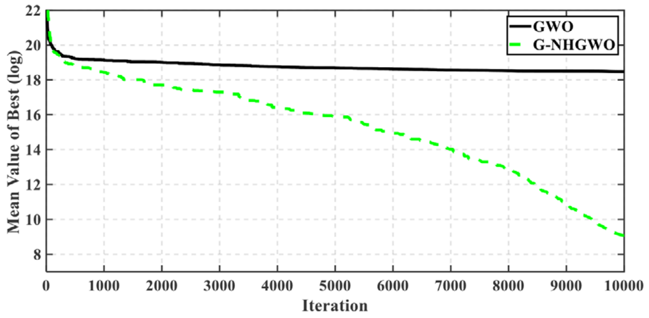
the original GWO and Winner means whether G-NHGWO outperforms GWO (+) or reaches worse solutions than those of (–).

According to the results given in this table, the proposed algorithm, G-NHGWO, was able to fix the defect of the original GWO algorithm and succeeded in optimizing a wide range of real-world shifted functions. The G-NHGWO algorithm outperformed the original GWO in 13 out of the 14 test functions and even reached much better solutions for test functions, such as F1 and F6. The proposed algorithm only performs

1350911x, 2021, 13, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/ell2.12176, Wiley Online Library on [26/02/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



a



b

Fig. 1 *The convergence characteristics of G-NHGWO and original GWO algorithms: (a) F1 test function; (b) F6 test function*

worse than the original algorithm for F11; however, the value of the objective function obtained by the proposed algorithm for this test function is not noticeably different compared to that of the original algorithm. Additionally, based on the %Imp parameter, in 10 out of 14 functions, G-NHGWO leads to a mean index that is more than 50% lower than that of GWO. Furthermore, the convergence characteristics of the algorithms for F1 and F6 are depicted in Figure 1, which clearly shows that the G-NHGWO algorithm has a better performance than the original GWO algorithm in escaping from the local optima and achieving better global solutions.

In the second part of the simulation studies, in order to compare the performance of the proposed G-NHGWO with that of GWO in solving a real-world engineering problem, they were used for solving the economic load dispatch (ELD) problem for 6-, 20-, and 40-generator test power systems [14–16]. The power transmission losses are considered in 6- and 20-generator test systems and the valve points effects are considered in 20-, and 40-generator test power systems [14–16]. The maximum number of objective function evaluations and the penalty factor for violating the power balance equation caused by generation deficit were set to 50,000 and 1×10^8 for all the studied test systems, respectively. Each algorithm was run 25 times for solving the ELD problem in each system and their statistical results were reported in Table 2. It is observed from this table that G-NHGWO also outperforms GWO in solving the real-world optimization problem. G-NHGWO yields a lower mean index and a much lower standard deviation among its final costs; this means that its output has a significantly fewer deviation from the optimal solution. Furthermore, it is seen that the best generation cost among all runs for each test system has been obtained by G-NHGWO. Table 3 presents the best solutions found by GWO and G-NHGWO algorithms for all the test systems.

Conclusion: A greedy non-Hierarchical grey wolf optimization (G-NHGWO) algorithm is proposed to enhance the optimization power of the original GWO algorithm for real-world shifted functions with non-zero optimal solutions. In the proposed algorithm, the social hierarchy of the grey wolves is neglected and three random wolves replace the three best wolves in the original GWO to guide the wolf pack in the hunting process. Furthermore, the update equations of the original GWO were modified to use the personal best positions of the grey wolves instead of their positions. Exploiting this strategy, the G-NHGWO algorithm has been able to fix the two main disadvantages of the original GWO algorithm: getting stuck in the local optimal solutions; and lack of population diversity. The obtained results on 14 real-world shift test functions prove the effectiveness and efficiency of the proposed G-NHGWO algorithm in optimizing real-world functions. There are numerous improved and hybrid versions of the original GWO algorithm. Since the proposed G-NHGWO is a basic version, most of the improvements and hybridisation techniques can be converted to their non-hierarchal counterparts to improve the performance further, which is the subject of future studies.

© 2021 The Authors. *Electronics Letters* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

Received: *12 January 2021* Accepted: *28 March 2021*
doi: 10.1049/ell2.12176

REFERENCES

1 Mirjalili, S., et al.: Grey wolf optimizer. *Adv. Eng. Softw.* **69**, 46–61 (2014). https://doi.org/10.1016/J.ADVENGSOFT.2013.12.007

2 Venkatakrishnan, G.R., et al.: Grey wolf optimizer to real power dispatch with non-linear constraints. *Comput. Model. Eng. Sci.* **115**(1), 25–45 (2018). https://doi.org/10.3970/cmes.2018.115.025

3 Sulaiman, M.H., et al.: Using the gray wolf optimizer for solving optimal reactive power dispatch problem. *Appl. Soft Comput.* **32**, 286–292 (2015). https://doi.org/10.1016/j.asoc.2015.03.041

4 Emary, E., et al.: Multi-objective gray-wolf optimization for attribute reduction. *Procedia Comput. Sci.* **65**, 623–632 (2015). https://doi.org/10.1016/j.procs.2015.09.006

5 Madhiarasan, M., Deepa, S.N.: ELMAN neural network with modified grey wolf optimizer for enhanced wind speed forecasting. *Circuits Syst.* **7**(10), 2975 (2016)

6 Khairuzzaman, A.K.M., Chaudhury, S.: Multilevel thresholding using grey wolf optimizer for image segmentation. *Expert Syst. Appl.* **86**, 64–76 (2017)

7 Qais, M.H., et al.: A grey wolf optimizer for optimum parameters of multiple PI controllers of a grid-connected PMSG driven by variable speed wind turbine. *IEEE Access* **6**, 44120–44128 (2018). https://doi.org/10.1109/ACCESS.2018.2864303

8 Mahmoud, H.Y., et al.: Hybrid cuckoo search algorithm and grey wolf optimiser-based optimal control strategy for performance enhancement of HVDC-based offshore wind farms. *IET Gener. Transm. Distrib.* **14**(10), 1902–1911 (2020). https://doi.org/10.1049/iet-gtd.2019.0801

9 Niu, P., et al.: The defect of the grey wolf optimization algorithm and its verification method. *Knowl Based Syst.* **171**, 37–43 (2019). https://doi.org/10.1016/j.knosys.2019.01.018

10 Eberhart, R., Kennedy, J.: A new optimizer using particle swarm theory. In: Proceedings of the Sixth International Symposium on Micro Machine and Human Science. IEEE, Nagoya (1995). https://doi.org/10.1109/mhs.1995.494215

11 Kumar, S., Chaturvedi, D.K.K.: Optimal power flow solution using fuzzy evolutionary and swarm optimization. *Int. J. Electr. Power Energy Syst.* **47**, 416–423 (2013). https://doi.org/10.1016/j.ijepes.2012.11.019

12 Suganthan, P.N., et al.: Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization. *KanGAL Rep.* **2005005**, (2005)

13 Ghasemi, M., et al.: New self-organising hierarchical PSO with jumping time-varying acceleration coefficients. *Electron. Lett.* **53**(20), 1360–1362 (2017). https://doi.org/10.1049/el.2017.2112

14 Abdullah, M.N., et al.: Modified particle swarm optimization for economic-emission load dispatch of power system operation. *Turkish J. Electr. Eng. Comput. Sci.* **23**(Sup. 1), 2304–2318 (2015)

15 Su, C.T., Lin, C.T.: New approach with a Hopfield modeling framework to economic dispatch. *IEEE Trans. Power Syst.* **15**(2), 541–545, 2000. https://doi.org/10.1109/59.867138

16 Wang, L., Li, L.P.: An effective differential harmony search algorithm for the solving non-convex economic load dispatch problems. *Int. J. Electr. Power Energy Syst.* **44**(1), 832–843 (2013). https://doi.org/10.1016/j.ijepes.2012.08.021

1350911x, 2021, 13, Downloaded from https://ietresearch.onlinelibrary.wiley.com/doi/10.1049/ell2.12176, Wiley Online Library on [26/02/2025]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License