Grey Wolf Optimization (GWO) Algorithm

Chapter *in* Studies in Computational Intelligence · July 2018

DOI: 10.1007/978-981-10-5221-7_9

3 authors:

Hossein Rezaei
University of Tehran

Omid Bozorg-Haddad
University of Tehran

Xuefeng Chu
North Dakota State University

Chapter 9
Grey Wolf Optimization (GWO) Algorithm

Hossein Rezaei, Omid Bozorg-Haddad and Xuefeng Chu

Abstract This chapter describes the grey wolf optimization (GWO) algorithm as one of the new meta-heuristic algorithms. First, a brief literature review is presented and then the natural process of the GWO algorithm is described. Also, the optimization process and a pseudo code of the GWO algorithm are presented in this chapter.

9.1 Introduction

Grey wolf optimization (GWO) is one of the new meta-heuristic optimization algorithms, which was introduced by Mirjalili et al. (2014). Gholizadeh (2015) developed the GWO algorithm to solve an optimization problem of double-layer grids considering the nonlinear behavior. The results illustrated that GWO had a better performance than other algorithms in finding the optimal design of nonlinear double-layer grids. Mirjalili (2015) used the GWO algorithm to learn multi-layer perceptron (MLP) for the first time. In the study, the results of GWO were compared with those from particle swarm optimization (PSO), genetic algorithm (GA), ant colony optimization (ACO), and evolution strategy (EA), and indicated the higher performance of GWO. Saremi et al. (2015) coupled GWO with the evolutionary population dynamic (EPD) to improve the performance of the basic GWO

H. Rezaei · O. Bozorg-Haddad (✉)
Department of Irrigation and Reclamation Engineering, Faculty of Agricultural Engineering and Technology, College of Agriculture and Natural Resources, University of Tehran, 31587-77871 Karaj, Tehran, Iran
e-mail: OBHaddad@ut.ac.ir

H. Rezaei
e-mail: HosseinRezaie18@ut.ac.ir

X. Chu
Department of Civil and Environmental Engineering, North Dakota State University, Dept 2470, Fargo, ND 58108-6050, USA
e-mail: Xuefeng.Chu@ndsu.edu

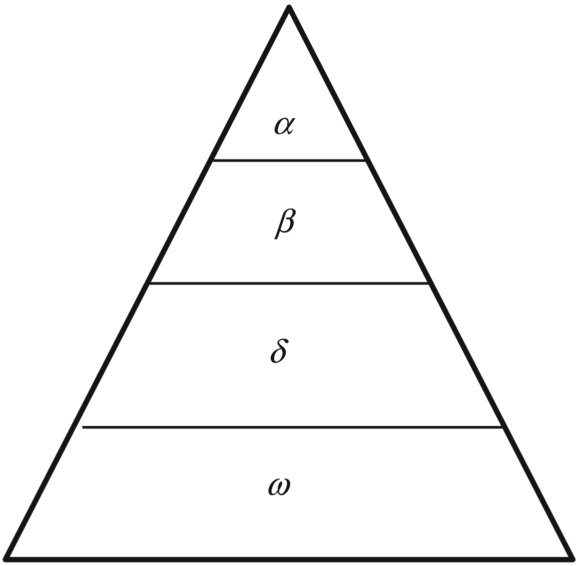
O. Bozorg-Haddad (ed.), *Advanced Optimization by Nature-Inspired Algorithms*, Studies in Computational Intelligence 720, DOI 10.1007/978-981-10-5221-7_9

algorithm by removing weak individuals from the society. Comparison with the basic GWO illustrated that the proposed algorithm had a better performance in conversion rate and exploration, and also avoided trapping into local optima. Sulaiman et al. (2015) used GWO to solve an optimal reactive power dispatch (ORPD) problem and compared with swarm intelligence (SI), evolutionary computation (EC), PSO, harmony search algorithm (HAS), gravity search algorithm (GSA), invasive weed optimization, and modified imperialist competitive algorithm with invasive weed optimization (MICA-IWO). The results demonstrated that GWO had more desirable optimal solution than others.

9.2 Natural Process of the GWO Algorithm

GWO is inspired by social hierarchy and the intelligent hunting method of grey wolves. Usually, grey wolves are at the top of the food chain in their life areas. Grey wolves mostly live in a pack of 5–12 individuals. In particular, in grey wolves' life there is a strict social hierarchy. As shown in Fig. 9.1, the leaders of a pack of grey wolves (alpha) are a male and female wolves that often are responsible for making decisions for their pack such as sleep place, hunting, and wake-up time. Mostly, other individuals of the pack must obey the decision made by alpha. However, some democratic behaviors in the social hierarchy of grey wolves can be observed (alpha may follow other individuals of the pack). In gatherings, individuals confirm the alpha's decision by holding their tails down. It is also interesting to know that it

Fig. 9.1 Social hierarchy of grey wolves



is not necessary for the alpha to be the strongest ones in the pack. Managing the pack is the main role of the alpha. In a pack of grey wolves, discipline and organization are the most important. The level next to alpha in the social hierarchy of grey wolves is beta and the role of beta is to help alpha in making decisions. Beta can be either male or female wolves and beta can be the best candidate of substitution for alpha when one of them becomes old or dies. The beta must respect alpha, but he/she can command other individuals. Beta is the consultant of alpha and responsible for disciplining the pack. The beta reinforces the orders of alpha and gives alpha the feedbacks. The weakest level in a pack of grey wolves is omega that plays a role of scapegoat. The wolves at the level of omega have to obey other individuals' orders and they are the last wolves that are allowed to eat food. Omega seems to be the least important individuals in the pack, but without omega, internal fight and other problems can be observed. This can be attributed to the omega's venting role of violence and frustration of other wolves, which helps satisfy other individuals and maintain the dominant structure of grey wolves. Sometimes, omega plays the role of babysitter in the pack. The remaining wolves, other than alpha, beta, and omega, are called subordinate (delta). The wolves at the level of delta obey the alpha and beta wolves and dominate the omega wolves. They act as scouts, sentinels, elders, hunters, and caretakers in the pack. Scouts are responsible for looking after boundaries and territory and also they should alarm the pack in facing to danger. Sentinels are in charge of security establishment. Elders are the experienced wolves that are candidates for alpha and beta. Hunters help alpha and beta in hunting and preparing food for the pack, while caretakers should look after the weak, ill, and wounded wolves.

In addition to the social hierarchy in a pack of grey wolves, group hunting is one of the interesting social behaviors of grey wolves too. According to Muro et al. (2011) grey wolves' hunting includes the following three main parts:

(1) Tracking, chasing, and approaching the prey.
(2) Pursuing, encircling, and harassing the prey till it stops moving.
(3) Attacking the prey.

These two social behaviors of grey wolves' pack (social hierarchy and hunting technique) are modeled in the GWO algorithm.

9.3 Mathematical Model of the GWO Algorithm

In this section, mathematical modeling of the social hierarchy of grey wolves, and their hunting technique (tracking, encircling, and attacking prey) in the GWO algorithm is detailed.

9.3.1 Social Hierarchy

In order to mathematically model the social hierarchy of grey wolves in the GWO algorithm, the best solution is considered as alpha (α). Therefore, the second and third best solutions are respectively considered as beta (β) and delta (δ), and other solution is assumed to be omega (ω). In the GWO algorithm, hunting (optimization) is guided by α, β, and δ, and ω wolves follow them.

9.3.2 Encircling the Prey

As aforementioned, grey wolves in the process of hunting, encircle the prey. The grey wolves' encircling behavior to hunt for a prey can be expressed as (Mirjalili et al. 2014):

$$\vec{D} = \left|\vec{C}.\vec{X}_p(t) - \vec{X}(t)\right| \tag{9.1}$$

$$\vec{X}(t+1) = \vec{X}_p(t) - \vec{A}.\vec{D} \tag{9.2}$$

where t = iteration number; $\vec{A}$ and $\vec{C}$ = coefficient vectors; $\vec{X}_P$ = vector of the prey's positions; $\vec{X}$ = vector of the grey wolf's positions; and $\vec{D}$ = calculated vector used to specify a new position of the grey wolf. $\vec{A}$ and $\vec{C}$ can be calculated by Mirjalili et al. (2014):

$$\vec{A} = 2\vec{a}.\vec{r_1} - \vec{a} \tag{9.3}$$

$$\vec{C} = 2.\vec{r_2} \tag{9.4}$$

where $\vec{a}$ = vector set to decrease linearly from 2 to 0 over the iterations; and $\vec{r_1}$ and $\vec{r}_2$ = random vectors in [0,1]. As shown in Fig. 9.2, a grey wolf at (x, y) can change its position based on the position of prey at (x', y'). Different places to the best agent can be achieved with respect to the current position by regulating the $\vec{A}$ and $\vec{C}$. For instance, by setting $\vec{A} = [1, 0]$ and $\vec{C} = [1, 1]$, the position of the grey wolf is updated to $(x' - x, y')$.

Note that the random $\vec{r_1}$ and $\vec{r_2}$ vectors let the grey wolf select any positions/nodes in Fig. 9.2. Therefore, a grey wolf can be placed in each random position around the prey that is calculated by using Eqs. (9.1) and (9.2). Following the same way, in an n-dimensional decision space grey wolves can move to any nodes of a hypercube around the best solution (position of the prey). They can distinguish the position of the prey from others and encircle it. Usually, hunting operation is guided by α, and β and δ provide support for α. In a decision space of an optimization problem we do not have any idea about the optimum solution.

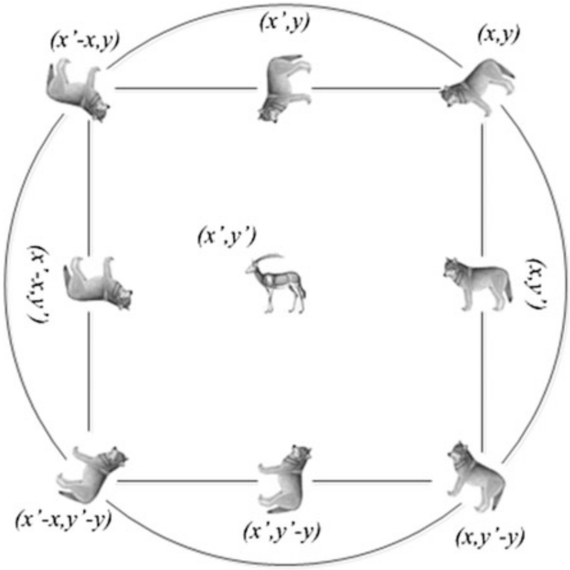


Fig. 9.2 Attacking toward prey versus searching for prey

Thus, in order to simulate the hunting behavior of grey wolves, we assume that α (best candidate for the solution), β, and δ have more knowledge about the potential position of the prey. Therefore, the algorithm saves three best solutions achieved so far and forces others (i.e., omega wolves) to update their positions to achieve the best place in the decision space. In the optimization algorithm, such a hunting behavior can be modeled by Mirjalili et al. (2014):

$$\vec{D}_{\alpha} = \left|\vec{C}_1.\vec{X}_a - \vec{X}\right|,\ \vec{D}_{\beta} = \left|\vec{C}_2.\vec{X}_{\beta} - \vec{X}\right|,\ \vec{D}_{\delta} = \left|\vec{C}_3.\vec{X}_{\delta} - \vec{X}\right| \tag{9.5}$$

$$\vec{X}_1 = \vec{X}_{\alpha} - A_1.(\vec{D}_{\alpha}),\ \vec{X}_2 = \vec{X}_{\beta} - A_2.(\vec{D}_{\beta}),\ \vec{X}_3 = \vec{X}_{\delta} - A_2.(\vec{D}_{\delta}) \tag{9.6}$$

$$\vec{X}_1 = \vec{X}_{\alpha} - A_1.(\vec{D}_{\alpha}) \tag{9.7}$$

Figure 9.3 shows how the search agent updates the positions of α, β, and δ in a 2D search space. As shown in Fig. 9.3, the final position (solution) is inside a circle that is specified based on the positions of α, β, and δ in the decision space. In other words, α, β, and δ estimate the positions of prey and other wolves and then update their new positions, randomly around the prey.

9.3.3 Attacking the Prey

As aforementioned, grey wolves finish the hunting process by attacking the prey until it stops moving. In order to model the attacking process, the value of $\vec{a}$ can be

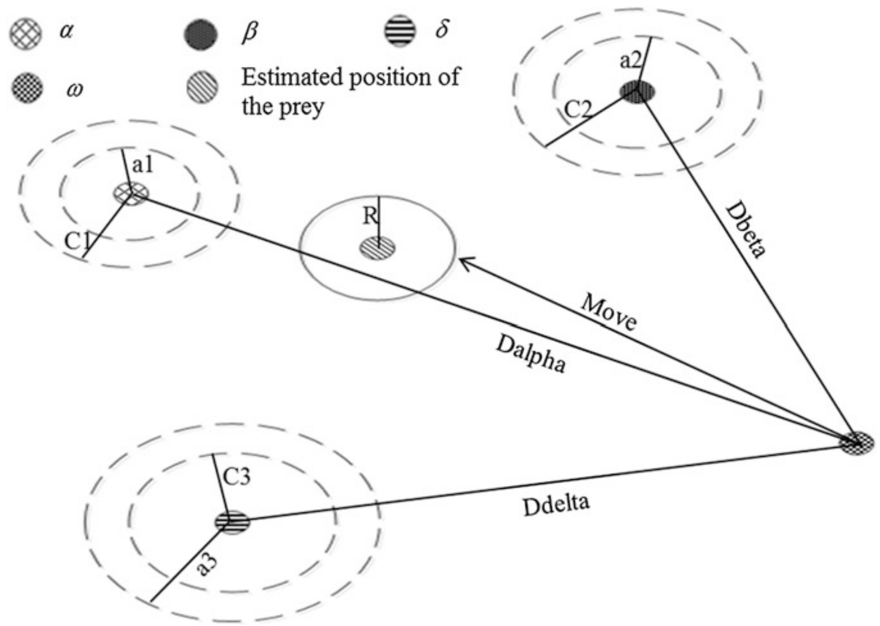


Fig. 9.3 Updating of positions in the GWO algorithm

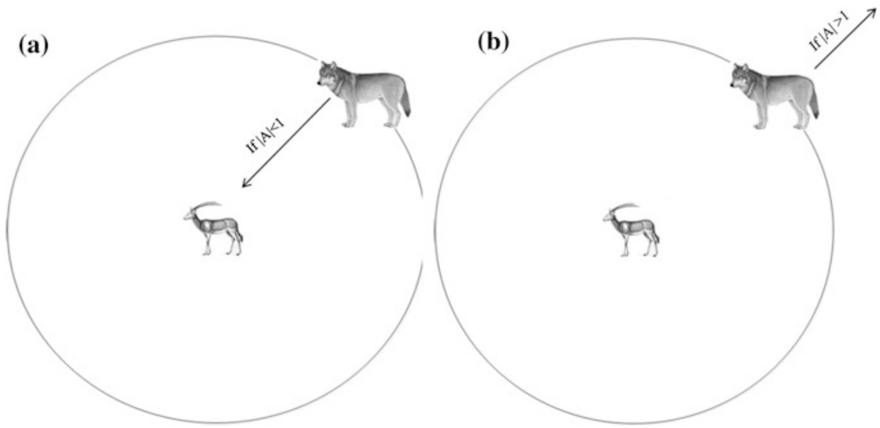


Fig. 9.4 Attacking toward prey and searching for prey

decreased in different iterations. Note that as $\overrightarrow{a}$ decreases the fluctuation rate of $\overrightarrow{A}$ decreases too. In other words, $\overrightarrow{A}$ is a random value in the range of $[-2a, 2a]$ where a decreases from 2 to 0 over iterations. When the random value of $\overrightarrow{A}$ is being in the range of $[-1, 1]$. The next position of a wolf can be between the current position and the prey position. As illustrated in Fig. 9.4, when $|A| < 1$ grey wolves will attack the prey.

By using the operators provided so far, the GWO algorithm lets the search agent to update its position based on the positions of α, β, and δ(move toward the prey). It is true that the encircling process provided as an operator in the GWO algorithm limits the solutions around local optima, but GWO also has many other operators to discover new solutions.

9.3.4 Searching for the Prey (Exploration)

Grey wolves often search for the prey according to the positions of α, β, and δ. They diverge from each other to explore the position of prey and then converge to attack the prey. In order to mathematically model the divergence of grey wolves, $\vec{A}$ can be utilized. $\vec{A}$ is a random vector that is greater than 1 or less than −1 to force the search agent to diverge from the prey, which emphasizes the global search in GWO. Figure 9.4 illustrates that when $\left|\vec{A}\right| > 1$, the grey wolf is forced to move away from the prey (local optimum) to search for better solutions in the decision space.

The GWO algorithm has another component $\left(\vec{C}\right)$ that assists the algorithm to discover new solutions. As shown in Eq. (9.4), the elements of vector $\vec{C}$ are within the range of $[0, 2]$. This component provides random weights for the prey to randomly emphasize $(C > 1)$ or deemphasize $(C < 1)$ the impact of the prey in defining the distance in Eq. (9.1). This component helps the GWO algorithm to behave more randomly and in favor of exploration, and keep the search agent away from local optima during the optimization process. Note that unlike A, C decreases nonlinearly. C is required in the GWO algorithm because not only in the initial iteration but also in the final iteration, it provides a global search in the decision space. This component is very useful in avoidance of local optima, especially in the

Table 9.1 Characteristics of the GWO algorithm

General algorithm	Grey wolf optimization algorithm
Decision variable	Grey wolf
Solution	Position of grey wolf
Old solution	Old position of grey wolf
New solution	New position of grey wolf
Best solution	Position of prey
Fitness function	Distance between grey wolf and prey
Initial solution	Initial random position of grey wolf
Selection	–
Process of generating new solution	Hunting operators (encircling and attacking prey)

final iteration. The C vector can be used as a hedge of approaching the prey in nature. Generally, the hedge can be seen in a nature hunting process of grey wolves. This hunting technique prevents grey wolves from quickly approaching the prey (this is truly what C does in the optimization process of the GWO algorithm). Table 9.1 presents the characteristics of the GWO algorithm.

9.4 Optimization Process in GWO Algorithm

The optimization process of GWO starts with creating random population of grey wolves (candidate solutions). Over the iterations, α, β, and δ wolves estimate the probable position of the prey (optimum solution). Grey wolves update their positions based on their distances from the prey. In order to emphasize exploration and exploitation during the search process, parameter a should decrease from 2 to 0. If $\left|\vec{A}\right| > 1$, the candidate solutions diverge from the prey; and if $|A| < 1$, the candidate solutions converge to the prey. This process continues and the GWO algorithm is terminated if the stopping criteria are satisfied. To understand how the GWO algorithm solves optimization problems theatrically, some notes can be summarized as follows:

- The concept of social hierarchy in the GWO algorithm helps grade the solutions and save the best solutions up to the current iteration.
- The encircling mechanism defines a 2D circle-shaped neighbor and the solution (in higher dimensions, the 2D circle can be extended to a 3D hyper-sphere).
- The random parameters (A and C) help grey wolves (candidate solutions) to define different hyper-spheres with random radii.
- The hunting approach implemented in the GWO algorithm allows grey wolves (candidate solutions) to locate the probable position of the prey (optimum solution).
- The adaptive values of parameters A and a guarantee exploration and exploitation in the GWO algorithm and also allow it to easily transfer between exploration and exploitation.
- By decreasing the values of A, a half of iterations are assigned to exploration $\left(\left|\vec{A}\right| > 1\right)$ and the other half of iterations are assigned to exploitation $(|A| < 1)$.
- a and C are two main parameters of the GWO algorithm.

Figure 9.5 shows the flowchart of the GWO algorithm with details on the optimization process.

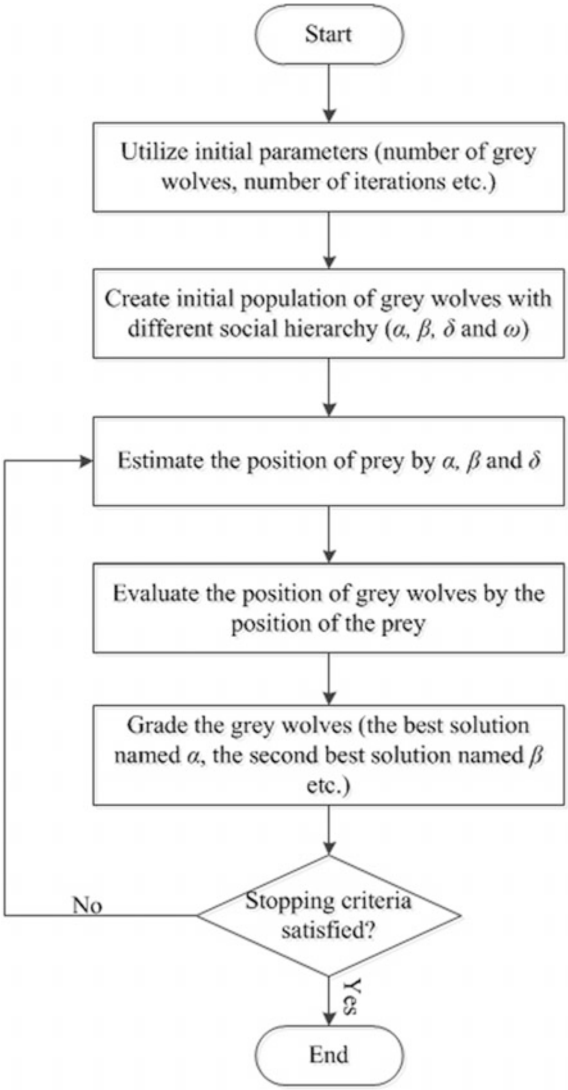


Fig. 9.5 Flowchart of the GWO algorithm

9.5 Pseudocode of GWO

Begin

Initialize the population of grey wolves X_i ($i = 1, 2, \ldots, n$)

Initialize a, A, and C

Calculate the fitness values of search agents and grade them. (X_α= the best solution in the search agent, X_β= the second best solution in the search agent, and X_δ= the third best solution in the search agent)

t= 0

While (t < Max number of iterations)

For each search agent

Update the position of the current search agent by Equation (9.7)

End for

Update a, A, and C

Calculate the fitness values of all search agents and grade them

Update the positions of X_α, X_β, and X_δ

t= t+1

End while

End

9.6 Conclusions

This chapter described the grey wolf optimization (GWO) algorithm as one of the new meta-heuristic algorithms. The GWO algorithm was inspired by the lift style of the pack of grey wolves (social hierarchy and hunting mechanism). Also, this chapter presented a brief literature review of GWO, described the natural process of grey wolves' life style and the mathematical equations of GWO, and finally presented a pseudocode of GWO.

References

Gholizadeh, S. (2015). Optimal design of double layer grids considering nonlinear behaviour by sequential grey wolf algorithm. *Journal of Optimization in Civil Engineering, 5*(4), 511–523.

Mech, L. D. (1999). Alpha status, dominance, and division of labor in wolf packs. *Canadian Journal of Zoology, 77*(8), 1196–1203.

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software, 69*(2014), 46–61.

Mirjalili, S. (2015). How effective is the grey wolf optimizer in training multi-layer perceptron. *Applied Intelligence, 43*(1), 150–161.

Mirjalili, S. M., & Mirjalili, S. Z. (2015). Full optimizer for designing photonic crystal waveguides: IMoMIR framework. *IEEE Photonics Technology Letters, 27*(16), 1776–1779.

Mirjalili, S. M., Mirjalili, S., & Mirjalili, S. Z. (2015). How to design photonic crystal LEDs with artificial intelligence techniques. *Electronics Letters, 51*(18), 1437–1439.

Muro, C., Escobedo, R., Spector, L., & Coppinger, R. (2011). Wolf-pack (Canis Lupus) hunting strategies emerge from simple rules in computational simulations. *Behavioral Processes, 88*(3), 192–197.

Naderizadeh, M., & Baygi, S. J. M. (2015). Statcom with grey wolf optimizer algorithm based pi controller for a grid Connected wind energy system. *International Research Journal of Applied and Basic Sciences, 9*(8), 14–21.

Noshadi, A., Shi, J., Lee, W. S., Shi, P., & Kalam, A. (2015). Optimal PID-type fuzzy logic controller for a multi-input multi-output active magnetic bearing system. *Neural Computing and Applications, 27*(7), 1–16.

Saremi, S., Mirjalili, S. Z., & Mirjalili, S. M. (2015). Evolutionary population dynamics and grey wolf optimizer. *Neural Computing and Applications, 26*(5), 1257–1263.

Sulaiman, M. H., Mustaffa, Z., Mohamed, M. R., & Aliman, O. (2015). Using the grey wolf optimizer for solving optimal reactive power dispatch problem. *Applied Soft Computing, 32* (2015), 286–292.

Wong, L. I., Sulaiman, M. H., & Mohamed, M. R. (2015). Solving economic dispatch problems with practical constraints utilizing grey wolf optimizer. *Applied Mechanics and Materials, 785* (2015), 511–515. Trans Tech Publications.

Yusof, Y., & Mustaffa, Z. (2015). Time series forecasting of energy commodity using grey wolf optimizer. In *Proceedings of the international multiconference of engineers and computer scientists (IMECS 2015)*, Hong Kong, 18–20 March.