GREY WOLF OPTIMIZER(GWO)

Dr. Rajesh Kumar
Associate Professor

ELECTRICAL ENGINEERING

MNIT JAIPUR

April 17, 2015

OUTLINE

About Grey Wolf

- Wolf is characterised by power full teeth, bushy tail and lives and hunts in packs. The average group size is 5-12.
- Their natural habitats are found in the mountains, forests, plains of North America, Asia and Europe.
- Grey wolf (Canis lupus) belongs to Canidae family.
- Grey wolves are considered as apex predators, meaning that they are at the top of the food chain.

Developers of Algorithm

Seyedali Mirjalili

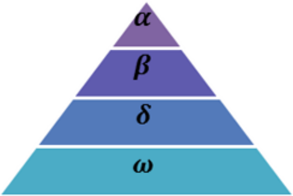
Seyed Mohammad Mirjalili

Andrew Lewis

Wolf behaviour in nature

Social behaviour

- Hierarchy exits in pack
- α is the leader and decision maker.
- β and δ assist α in decision making.
- Rest of the wolves (ω) are followers.



Wolf behaviour in nature

Hunting behaviour

Group hunting behaviour is of equal interest in studying optimization.

- Tracking, chasing, and approaching the prey.
- Pursuing, encircling, and harassing the prey until it stops moving.
- Attacking the prey.

Approach, track and pursuit

Pursuit

Harass

Encircling

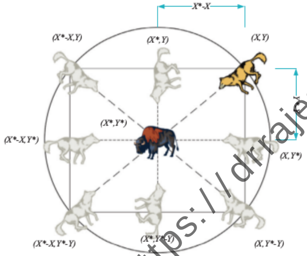
At the end, when the prey stops, wolves make a approximate regular polygon around it and lay down

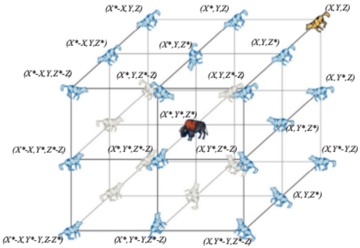
Algorithm development

Social hierarchy

In order to mathematically model the social hierarchy of wolves when designing GWO, we consider the fittest solution as the alpha (α). Consequently, the second and third best solutions are named beta (β) and delta (δ) respectively. The rest of the candidate solutions are assumed to be omega (ω). In the GWO algorithm the hunting (optimization) is guided by α , β, and δ. The ω wolves follow these three wolves.

Encircling prey





Encircling Prey: Mathematical Modeling

The mathematical model of the encircling behaviour is represented by the equations:

$$D = |CX_p - AX(t)| \tag{1}$$

$$X(t+1) = X_p(t) - AD \tag{2}$$

Encircling Prey: Mathematical Modeling

- A and C are coefficient vectors given by:

$$A = 2ar_1a \tag{3}$$

$$C = 2r2 \tag{4}$$

- t is the current iteration
- X is the position vector of a wolf
- r_1 and r_2 are random vectors $\in [0, 1]$ and a linearly varies from 2 to 0
- More description in later slides

Hunting

- Grey wolves have the ability to recognize the location of prey and encircle them.
- The hunt is usually guided by the alpha. The beta and delta might also participate in hunting occasionally.
- However, in an abstract search space we have no idea about the location of the optimum (prey).
- In order to mathematically simulate the hunting behaviour, we suppose that the alpha, beta and delta have better knowledge about the potential location of prey.

Hunting

$$\vec{D}_{\alpha} = |\vec{C}_1.\vec{X}_{\alpha}(t) - \vec{X}(t)|, \vec{D}_{\beta} = |\vec{C}_2.\vec{X}_{\beta}(t) - \vec{X}(t)|, \vec{D}_{\delta} = |\vec{C}_3.\vec{X}_{\delta}(t) - \vec{X}(t)| \quad (5)$$

$$\vec{X}_1 = \vec{X}_{\alpha}(t) - \vec{A}_1.(\vec{D}_{\alpha}), \vec{X}_2 = \vec{X}_{\beta}(t) - \vec{A}_2.(\vec{D}_{\beta}), \vec{X}_3 = \vec{X}_{\delta}(t) - \vec{A}_3.(\vec{D}_{\delta}) \quad (6)$$

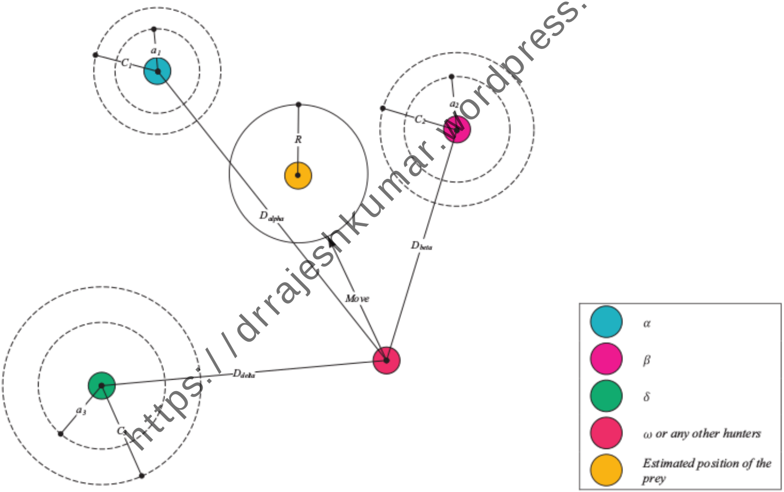
$$\vec{X}(t+1) = \frac{X_1 + X_2 + X_3}{3} \quad (7)$$

where t indicates the current iteration, $\vec{X}_{\alpha}(t)$, $\vec{X}_{\beta}(t)$ and $\vec{X}_{\delta}(t)$ are the position of the gray wolves α, β and δ at $t^t h$ iteration, $\vec{X}(t)$ presents the position of the gray wolf at t^{th} iteration.

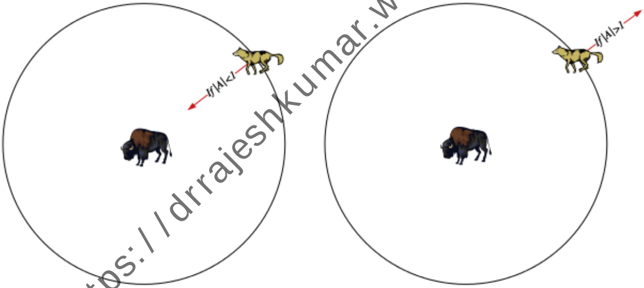
$$\vec{A}_{(.)} = 2\vec{a}.rand(0,1) - \vec{a} \quad (8)$$

$$\vec{C}_{(.)} = 2.rand(0,1) \quad (9)$$

Where $\vec{a}$ is the linear value varies from 2 to 0 according to iteration. $\vec{A}_{(.)}$ and $\vec{C}_{(.)}$ are the coefficient vector of α, β and δ wolfs.


a_1
C_1
a_2
C_2
R
D_{alpha}
D_{beta}
Move
D_{delta}
a_3
C_3
α
β
δ
ω or any other hunters
Estimated position of the prey

Attacking prey & Search for prey



Example

minimization of Korn function

$$f_{(x_1,x_2)} = min\{(x_1 - 5)^2 + (x_2 - 2)^2\}$$

Iteration 1

	x_1	x_2	$f(x)$
1	6.1686	4.4100	7.1739
2	6.2104	4.0935	5.8479
3	7.4231	8.3880	46.6773
4	2.8950	0.8703	5.7074
5	6.1062	3.7275	4.2079
6	6.3458	3.2158	3.2896
7	7.5690	6.1457	23.7866
8	6.2471	4.0456	5.7397
9	6.9965	4.5846	10.6663
10	4.7372	3.3048	1.7717
11	4.8148	3.4931	2.2637
12	5.9444	3.4433	2.9751

	x_1	x_2	$f(x)$
α	4.7372	3.3048	1.7717
β	4.8148	3.4931	2.2637
δ	5.9444	3.4433	2.9751

Update process

$$\vec{D}_{\alpha} = |2.rand().[4.7372, 3.3048] - [6.1686, 4.4100]|$$

$$X_1 = [4.7372, 3.3048] - (2\vec{a}.rand(0,1) - \vec{a})\vec{D}_{\alpha}$$

$$\vec{D}_{\beta} = |2.rand().[4.8148, 3.4931] - [6.1686, 4.4100]|$$

$$X_2 = [4.8148, 3.4931] - (2\vec{a}.rand(0,1) - \vec{a})\vec{D}_{\beta}$$

$$\vec{D}_{\delta} = |2.rand().[5.9444, 3.4433] - [6.1686, 4.4100]|$$

$$X_3 = [5.9444, 3.4433] - (2\vec{a}.rand(0,1) - \vec{a})\vec{D}_{\delta}$$

$$\vec{X}(1,:) = \frac{X_1 + X_2 + X_3}{3} = [4.0487, 2.6051]$$

$$\vec{a} = 2 - 2.(\frac{itr}{maxitr})$$

$$\vec{a} = 2 - 2.(\frac{1}{3})$$

$$\vec{a} = 1.3333$$

Iteration 2

	x_1	x_2	$f(x)$
1	4.0487	2.6051	1.2710
2	4.6492	3.0427	1.2103
3	5.4633	3.6633	2.9813
4	5.6096	3.5901	2.9001
5	4.6582	3.0302	1.1781
6	4.7476	3.3369	1.8509
7	4.2452	2.6600	1.0054
8	4.9026	3.2497	1.5712
9	4.5202	2.9588	1.1495
10	5.3971	3.5432	2.5392
11	4.1136	2.5382	1.0754
12	5.0927	3.1546	1.3418

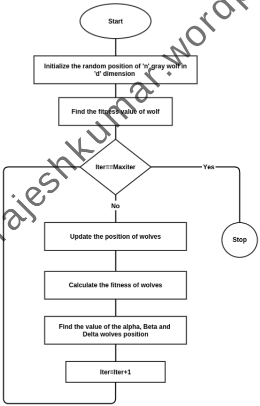
	x_1	x_2	$f(x)$
α	4.2452	2.6600	1.0054
β	4.1136	2.5382	1.0754
δ	5.0927	3.1546	1.3418

Iteration 3

	x_1	x_2	$f(x)$
1	4.4838	2.7843	0.8816
2	4.5634	2.8257	0.8725
3	4.5899	2.8395	0.8730
4	4.7486	2.9400	0.9467
5	4.6340	2.8684	0.8881
6	4.5957	2.8445	0.8767
7	4.5830	2.8366	0.8738
8	4.5787	2.8339	0.8729
9	4.5750	2.8321	0.8730
10	4.5724	2.8306	0.8727
11	4.5703	2.8295	0.8727
12	4.5696	2.8291	0.8727

	x_1	x_2	$f(x)$
α	4.5634	2.8257	0.8725
β	4.5696	2.8291	0.8727
δ	4.5750	2.8321	0.8730

Flow chart


Start
Initialize the random position of 'n' gray wolf in 'd' dimension
Find the fitness value of wolf
Iter==Maxiter
Yes
No
Stop
Update the position of wolves
Calculate the fitness of wolves
Find the value of the alpha, Beta and Delta wolves position
Iter=Iter+1

Advantages over other techniques

- Easy to implement due to simple structure.
- Less storage requirement than the other techniques.
- Convergence is faster due to continuous reduction of search space and Decision variables are very less (α, β and δ).
- It avoids local optima when applied to composite functions also.
- only two main parameters to be adjusted (a and C).

Unit Commitment Problem

- Unit Commitment (UC) is a very significant optimization task, which plays an important role in the operation planning of power systems.
- UCP is considered as two linked optimization decision processes, namely the unit-scheduled problem that determines on/off status of generating units in each time period of planning horizon, and the economic load dispatch problem.
- UCP is a complex nonlinear, mixed-integer combinational optimization problem with 01 variables that represents on/off status.

Unit commitment problem

Table : Total costs of the BGWO method for test systems

No. of Unit	Best Cost ($)	Average Cost ($)	Worst Cost ($)	Std. Deviation	CPU Time (Sec)
10	563937.3	563976.6	564017.7	40.2	31.3
20	1124687.9	1124837.7	1124941.1	128.7	58.7
40	2248280.0	2248397.6	2248614.0	174.2	124.6
60	3367893.4	3367881.1	3367933.4	37.9	216.9
80	4492399.4	4492608.1	4492672.5	154.4	347.5
100	5612309.4	5612377.2	5612496.3	96.9	505.6

Performance Comparison

Table : Comparison With Other Algorithms

	10	20	40	60	80	100
LR	565825	1130660	2258503	3394066	4526022	5657277
GA	565825	1126243	2251911	3376625	4504933	5627437
EP	564551	1125494	2249093	3371611	4498479	5623885
MA	565827	1128192	2249589	3370820	4494214	5616314
GRASP	565825	1128160	2259340	3383184	4525934	5668870
LRPSO	565869	1128072	2251116	3376407	4496717	5623607
PSO	564212	1125983	2250012	3374174	4501538	5625376
IBPSO	563977	1125216	2248581	3367865	4491083	5610293
BFWA	563977	1124858	2248372	3367912	4492680	5612619
BGWO	563937.3	1124684.8	2248280.0	3367893.4	4492399.4	5612309.4

Thank You . . .

Mail: *rkumar.ee@gmail.com*