

Méthode des loups gris

DJIAVOUDINE QUTHBULHAMEED & AHMADI AMIR

Encadrant de projet : Boubaker DAACHI

8 décembre 2024



- 1 Introduction
- 2 L'état de l'art
- 3 Principe de GWO
- 4 Algorithme GWO
- 5 Methodes similaires
- 6 Exemple de démonstration
- 7 Conclusion



- Objectif
- Inspiration biologique
- Applications
- Avantages

L'état de l'art sur l'algorithme GWO

4/24

Origine

- Proposé par Mirjalili en 2014.
- Inspiré du comportement social des loups gris pour la chasse

Comparaisons

- Performances proches ou supérieures à PSO et GA dans certains cas.
- Simplicité et peu de paramètres à régler.
- Points forts : efficacité, simplicité, capacité à équilibrer exploration et exploitation.

L'état de l'art sur l'algorithme GWO (la suite)

5/24

Limites et évolutions

- Versions améliorées : GWO hybride, multi-objectif.
- Limites : risque de stagnation dans un optimum local, dépendance aux paramètres, Sensibilité à la façon dont les positions des loups sont initialisées.
- Applications : énergie, robotique, bioinformatique, ingénierie.

La nature et l'algorithme GWO

6/24

Étape dans la nature	Étape dans l'algorithme GWO
Hiérarchie sociale	Classement des solutions en $\alpha, \beta, \delta, \omega$
Encerclement de la proie	Mise à jour des positions en fonction de α, β, δ
Repérage de la proie	Évaluation de la fonction objectif pour identifier les leaders
Attaque de la proie	Exploitation des solutions proches
Exploration du territoire	Exploration de nouvelles régions
Collaboration et coordination	Ajustement collectif des solutions pour converger vers l'optimum

Hiérarchie Sociale

7/24

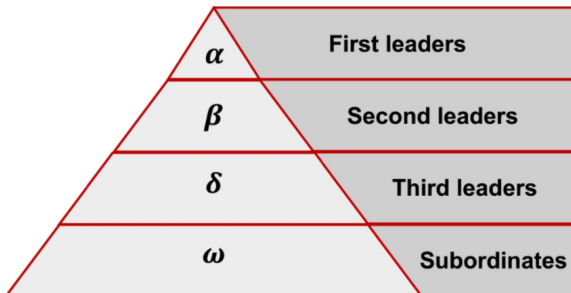
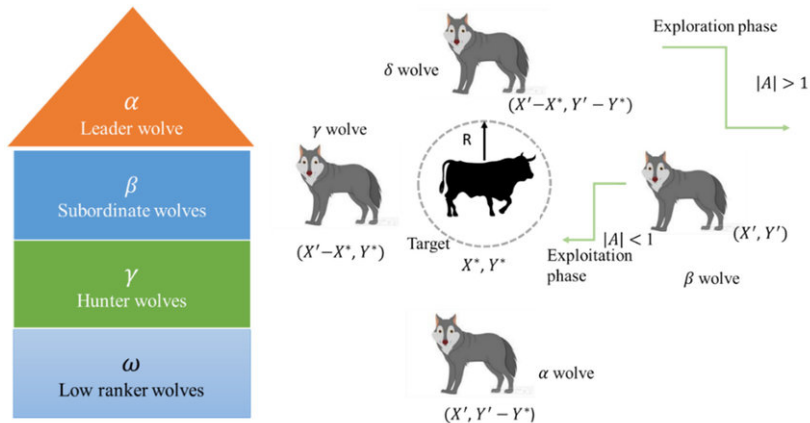


FIG. : Hiérarchie sociale du loup gris

Le loup alpha est considéré comme le loup dominant de la meute et tous ses ordres doivent être suivis par les membres de la meute.

Recherche et positionnement

8/24

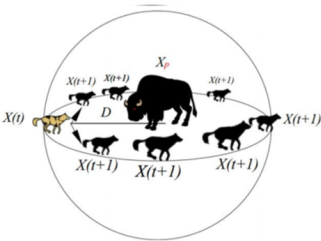


Les loups utilisent leurs sens (odorat, vue, ouïe) pour identifier une cible.

Ils analysent les informations collectives pour localiser la proie la plus accessible.

Encerclement de la proie

9/24



- Pendant la chasse, les loups gris adoptent une stratégie d'encerclement de leur proie.
- Ce comportement d'encerclement peut être modélisé à l'aide des équations mathématiques suivantes :

$$D = C \cdot X_p(t) - A \cdot X(t) \quad (1)$$

$$X(t+1) = X_p(t) - A \cdot D \quad (2)$$

- t représente l'**itération actuelle** dans le processus de chasse des loups gris.
- A et C sont des **vecteurs de coefficients** qui influencent les déplacements des loups et leur rapprochement avec la proie.
- X_p est le **vecteur position de la proie** et X indique le **vecteur position d'un loup gris**.

Encerclement de la proie (La suite)

10/24

- . **C** : Sert à savoir si les loups sont **proches** ou **éloignés** de la proie, avec des valeurs comprises entre [0, 1].
- . **A** : Définit le **déplacement des loups**.
 - Si **A est grand**, les loups se déplacent de manière plus large et imprécise
 - Si **A est petit**, les loups se déplacent de façon plus précise autour de la proie..

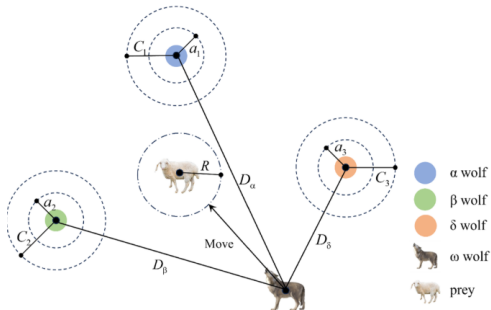
Les vecteurs **A** et **C** sont calculés à chaque itération selon les formules suivantes :

$$\mathbf{A} = 2\mathbf{a} \cdot \mathbf{r}_2 \cdot 2\mathbf{a} \quad (3)$$

$$\mathbf{C} = 2 \cdot \mathbf{r}_2 \quad (4)$$

Chasse au loup gris

11/24



Calcul des distances (D_α , D_β , D_δ) :

$$D_\alpha = C_1 \cdot X_\alpha - X$$

$$D_\beta = C_2 \cdot X_\beta - X$$

$$D_\delta = C_3 \cdot X_\delta - X$$

Mise à jour des positions (X_1 , X_2 , X_3) :

$$X_1 = X_\alpha - A_1 \cdot D_\alpha$$

$$X_2 = X_\beta - A_2 \cdot D_\beta$$

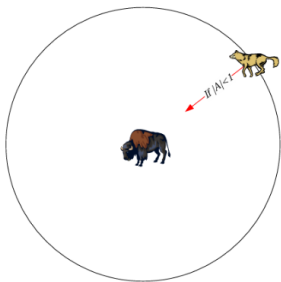
$$X_3 = X_\delta - A_3 \cdot D_\delta$$

Position finale ($X(t+1)$) :

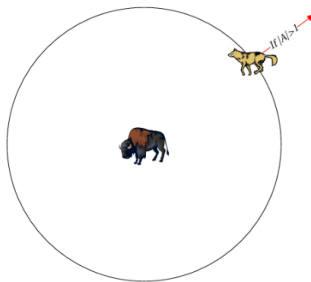
$$\frac{X_1 + X_2 + X_3}{3}$$

Attaque de la proie

12/24



- La valeur de **A** diminue progressivement de 2 à 0 au fil des itérations
- Cela réduit la plage de variation du coefficient A
- Lorsque **|A| < 1**, l'agent se rapproche de la proie en ajustant sa position



- **Lorsque $|A| > 1$**
- Les loups restent dans la phase de recherche
- Ils continuent à explorer d'autres zones pour localiser la proie

① Initialisation :

- Génération d'une population initiale de solutions (loups).
- Définition des coefficients de contrôle (\vec{A} , \vec{C} , a).

② Mise à jour des positions :

- Les positions sont ajustées en fonction des trois meilleurs loups (α, β, δ).

③ Encerclement et attaque :

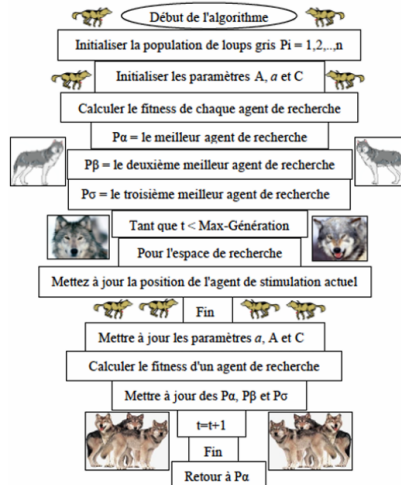
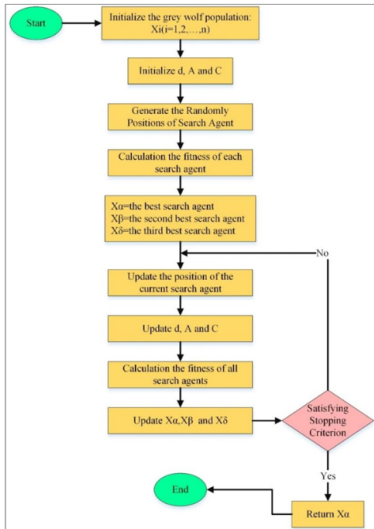
- L'équilibre entre exploration et exploitation est maintenu grâce aux coefficients \vec{A} et \vec{C} .

④ Critère d'arrêt :

- L'algorithme s'arrête lorsqu'un nombre maximal d'itérations est atteint ou si les solutions convergent.

L'algorithme des loups gris

14/24



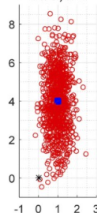
Schema simplifié

15/24

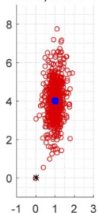
Initialiser la population des loups gris X_i ($i = 1, 2, \dots, n$)
 Initialiser les paramètres $a=2$, A et C (en utilisant l'équation 2 et 4)
 Calculer la fitness de chaque membre de la population
 $X\alpha$ = le meilleur agent de recherche
 $X\beta$ = le deuxième meilleur agent de recherche
 $X\delta$ = le troisième meilleur agent de recherche

Tant que ($t < \text{nombre maximum iterations}$)
 Pour chaque agent de recherche
 Mettre à jour la position de l'agent (en utilisant l'équation 6)
 Fin pour
 Mettre à jour a , A et C (en utilisant l'équation 2 et 4)
 Calculer la fitness de tous les agents de recherche
 Mettre à jour $X\alpha$, $X\beta$, $X\delta$

Fin tant que
 Retourner $X\alpha$

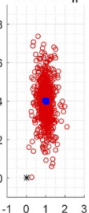
GWO, $a=2$ 

5-1

GWO, $a=2$ to 0

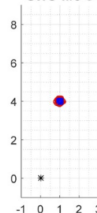
5-2

GWO w/o ||

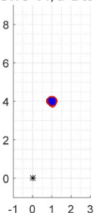


5-3

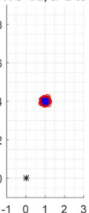
GWO w/o C



5-4

GWO-C1, $a=2$ to 0

5-5

GWO-C2, $a=2$ to 0

5-6

Comparaison avec d'autres algorithmes bio-inspirés

16/24

Caractéristique	GWO	CSA	ACO	Firefly Algorithm
Inspiration biologique	Hiérarchie sociale des loups gris	Comportement parasitaire du coucou	Comportement des fourmis dans la recherche de nourriture	Comportement des lucioles (attraction lumineuse)
Exploration	Basée sur α, β, δ	Lévy Flight pour des sauts aléatoires	Recherche basée sur les phéromones	Attraction vers des solutions plus brillantes
Exploitation	Exploitation dirigée par les leaders	Exploration locale avec Lévy Flight	Exploitation par le suivi des phéromones	Exploitation via l'attraction lumineuse
Méthode de convergence	Hiérarchie sociale et mise à jour des positions	Lévy Flights + sélection des meilleures solutions	Mise à jour des phéromones et évaporation	Attraction vers des solutions plus brillantes
Applications	Optimisation continue, multi-objectifs	Optimisation de fonctions complexes	Problèmes combinatoires (TSP, etc.)	Optimisation continue et conception de réseaux
Complexité	Modérée (dépend de la taille de la population)	Modérée (dépend de la taille des sauts)	Elevée (calculs de phéromones et évaporation)	Modérée (calculs d'attraction lumineuse)
Avantage	Bonne gestion de l'exploration/exploitation	Bonne gestion des solutions complexes	Efficace pour les problèmes combinatoires	Simple et efficace pour l'optimisation continue
Inconvénient	Sensible aux paramètres et aux conditions initiales	Convergence lente sur certains problèmes	Sensible à la quantité de phéromones	Moins efficace pour les problèmes combinatoires

- Ant Colony Optimization (ACO) 1992
Comportement des fourmis.
- Whale Optimization Algorithm (WOA) 2016
cet algorithme est basé sur le comportement de chasse des baleines à bosse.
Mieux adapté aux problèmes multidimensionnels.
- Cuckoo Search Algorithm (CSA) 2009
Comportement des coucous, qui parasitent les nids d'autres oiseaux.
- Artificial Bee Colony (ABC) 2005
Comportement des abeilles en recherche de nourriture.
il est excellente pour l'optimisation combinatoire.

1. Initialisation des positions des loups :

- Position du loup α : $X_\alpha = (5, 4)$
- Position du loup β : $X_\beta = (3, 3)$
- Position du loup δ : $X_\delta = (1, 2)$
- Position du loup à mettre à jour : $X = (2, 1)$

Les paramètres sont :

$$A = [0.5, -1.2], \quad C = [1.4, 1.1]$$

Intervalles des valeurs : $[0, 10]$

Démonstration (la suite)

19/24

2. Calcul des distances D_α , D_β , D_δ :Les distances D sont calculées :

$$D_\alpha = |C_1 \cdot X_\alpha - X|, \quad D_\beta = |C_2 \cdot X_\beta - X|, \quad D_\delta = |C_3 \cdot X_\delta - X|$$

● Pour D_α :

$$D_\alpha = |C \cdot X_\alpha - X| = |[1.4 \cdot 5, 1.1 \cdot 4] - [2, 1]|$$

$$D_\alpha = |[7.0, 4.4] - [2, 1]| = |[5.0, 3.4]| = [5.0, 3.4]$$

● Pour D_β :

$$D_\beta = |C \cdot X_\beta - X| = |[1.4 \cdot 3, 1.1 \cdot 3] - [2, 1]|$$

$$D_\beta = |[4.2, 3.3] - [2, 1]| = |[2.2, 2.3]| = [2.2, 2.3]$$

● Pour D_δ :

$$D_\delta = |C \cdot X_\delta - X| = |[1.4 \cdot 1, 1.1 \cdot 2] - [2, 1]|$$

$$D_\delta = |[1.4, 2.2] - [2, 1]| = |[-0.6, 1.2]| = [0.6, 1.2]$$

Démonstration (la suite)

20/24

3. Calcul des nouvelles positions $X_{(i+1)}$:

$$X_{(i+1)} = \frac{1}{3} (X_1 + X_2 + X_3)$$

où

$$X_1 = X - A \cdot D_\alpha, \quad X_2 = X - A \cdot D_\beta, \quad X_3 = X - A \cdot D_\delta$$

• Calcul de X_1 :

$$X_1 = X - A \cdot D_\alpha = [2, 1] - [0.5, -1.2] \cdot [5.0, 3.4]$$

$$X_1 = [2, 1] - [2.5, -4.08] = [-0.5, 5.08]$$

• Calcul de X_2 :

$$X_2 = X - A \cdot D_\beta = [2, 1] - [0.5, -1.2] \cdot [2.2, 2.3]$$

$$X_2 = [2, 1] - [1.1, -2.76] = [0.9, 3.76]$$

• Calcul de X_3 :

$$X_3 = X - A \cdot D_\delta = [2, 1] - [0.5, -1.2] \cdot [0.6, 1.2]$$

$$X_3 = [2, 1] - [0.3, -1.44] = [1.7, 2.44]$$

Nouvelle position $X_{(i+1)}$:

$$X_{(i+1)} = \frac{1}{3} ([-0.5, 5.08] + [0.9, 3.76] + [1.7, 2.44])$$

$$X_{(i+1)} = \frac{1}{3} [2.1, 11.28] = [0.7, 3.76]$$

4. Vérification des bornes : Les nouvelles positions doivent être dans l'intervalle $[0, 10]$.

Pour $X_{\text{nouveau}} = [0.7, 3.76]$, les valeurs sont dans $[0, 10]$.

Position finale après cette itération :

Le loup considéré se déplace vers la nouvelle position :

$$X_{(i+1)} = [0.7, 3.76]$$

Cela conclut une itération de l'algorithme GWO.

Conclusion

22/24

- Simplicité et son efficacité en font une méthode incontournable.
- Développement de versions hybrides pour surmonter ses limites (ex. GWO-PSO)
- Applications futures dans des domaines comme l'optimisation multi-objectifs et le deep learning.
- L'algorithme GWO, en s'inspirant de la nature, nous enseigne que la collaboration et l'adaptation sont des clés pour résoudre les problèmes les plus complexes.

Références

23/24

- ▶ Anurag ARNAB et al. « Vivit : A video vision transformer ». In : *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 2021, p. 6836-6846.

Merci pour votre attention