

Aide à la décision - R6.06

Optimisation

Déterminer le minimum d'une fonction

Optimisation par essais particulières

Michel Salomon

IUT Nord Franche-Comté
Département d'informatique

basé sur un support de Philippe Collard

Objectifs des systèmes bio-inspirés

- Créer des systèmes autonomes, robustes et évolutifs
- S'inspirer des systèmes naturels complexes
 - Animaux
 - Végétaux
 - Champignons, bactéries, virus, etc.
- Exemple : conception de systèmes de détection d'intrusions en s'inspirant des mécanismes des défenses immunitaires
- Caractéristiques
 - Auto-organisation
 - Adaptation
 - Distribution

Métaheuristiques distribuées (*Population-based*)

- Algorithmes évolutionnaires, de colonies de fourmis, etc.
- **Algorithme des essais particuliers**

Intelligence collective : conditions d'émergence

- Une information locale et limitée
 - Chaque individu ne possède qu'une connaissance partielle de l'environnement et n'a pas conscience de la totalité des éléments qui influencent le groupe
- Un ensemble de règles simples
 - Chaque individu obéit à un ensemble restreint de règles simples par rapport au comportement du système global
- Les interactions sont multiples
 - Chaque individu est en relation avec seulement un ou plusieurs autres individus du groupe
- La structure émergente est utile à la collectivité
 - Les individus trouvent un bénéfice à collaborer et leur performance est meilleure que s'ils avaient été seuls

Illustration avec les oiseaux migrateurs

- Les oiseaux migrateurs doivent parcourir de très longues distances, dans des conditions parfois difficiles
- Ils optimisent leur déplacement en terme d'énergie dépensée
 - Les oies sauvages adoptent des formations en V
 - Chaque oiseau prend l'aspiration de son prédécesseur
 - Permet d'étendre la distance de vol de près de 70%
 - Il y a un prix à payer
 - Un individu seul vole en moyenne 24% plus vite qu'une volée
 - Il y a donc une perte en vitesse

L'intelligence collective s'observe

- Animaux se déplaçant en formation (oiseaux migrateurs, bancs de poissons)
- Insectes sociaux (fourmis, termites et abeilles)

Points communs qui caractérisent l'intelligence collective

- Individus **grégaire**s car ils obtiennent un avantage substantiel à chasser, se déplacer ou vivre, en groupe
- Ils interagissent de manière **locale** par le moyen de signaux (grognements, phéromones, attitudes)
- L'individu seul répond instinctivement à certains stimuli
- La **coordination du groupe est implicite** et se fait au travers de règles comportementales très simples au niveau individuel

Métaheuristique d'optimisation proposée en 1995

Particle Swarm Optimization (PSO)

- Russel Eberhart (ingénieur en électricité)
- Jim Kennedy (socio-psychologue)

Principe

- Une population d'individus (des particules en l'occurrence)
- Des individus avec des capacités de perception, mémorisation et calcul limitées
- Dynamique induite par des interactions locales
- Observation de comportements globaux "émergents"

Collaboration plutôt que **compétition**

Essais particulières - utilisation

- Efficaces sur un espace de recherche **continu**
- Dynamiques définis sur un espace temps **discret**

Utilisés dans des domaines variés

- Régulation de systèmes électriques
- Conception d'ailes d'avions
- Analyse d'images
- etc.

Sources

- Maurice Clerc. L'optimisation par essais particulières, Hermès Science, 2005
- Kennedy, J., Eberhart, R.C. and Shi, Y. Swarm Intelligence, Morgan Kaufmann Publishers, 2001

État et comportement d'une particule

- Modèle social simplifié
 - Mime le comportement d'un(e) nuée d'oiseau/banc de poissons
 - Basé sur des règles qui permettent de soudain changement de direction, dispersion, regroupement, etc.
- État (instantané) d'une particule
 - Position : représente une solution du problème
 - Vitesse : direction pour un futur déplacement
- Comportement
 - Se déplacer dans l'espace de recherche
 - Le but est de se positionner sur des solutions optimales

Pas de mécanisme de sélection

- Une particule médiocre est conservée
- Succès en sortant des "sentiers battus"

Essais particuliers - une particule - 2/3

Mise à jour de la position d'une particule

Une particule détermine sa prochaine position en utilisant :

- Sa vitesse actuelle
- Sa meilleure performance
- La meilleure performance de ses voisines

Induit trois types de comportement...

- Égoïste : suivre sa propre voie
- Conservateur : revenir en arrière
- Panurgien : suivre le meilleur de tous

Compromis entre les trois types de comportement...

Compromis psycho-social entre confiance en soi
et influence de ses relations sociales

Déplacement d'une particule

Une combinaison pondérée des trois types de comportement

Prenant en compte :

- L'état instantané (position & vitesse) → **mémoire propre à court terme**
- La perf. personnelle → **mémoire propre à long terme**
- La perf. des voisins → **mémoire partagée**

Initialisation d'une particule

- 1 Fixer sa position aléatoirement dans l'espace de recherche
- 2 Fixer une vitesse aléatoirement
- 3 Définir son voisinage
 - Géographique (notion de distance) → dynamique
 - Social → fixé une fois pour toute au départ (Comment ?)

À chaque itération k (ou pas de temps) on fait bouger les particules

Équations pour calculer la nouvelle position d'une particule P_i

- Mettre à jour sa vitesse actuelle $V_i^k \Rightarrow V_i^{k+1}$

$$V_i^{k+1} = w \cdot V_i^k + c_1 \cdot (X_i^b - X_i^k) + c_2 \cdot (X_i^g - X_i^k) \quad (1)$$

avec :

- X_i^b sa meilleure position / solution
 - X_i^g la meilleure position / solution dans son voisinage
 - w est son inertie
 - c_1 et c_2 pondèrent l'attraction des meilleures solutions
- Mettre à jour sa position actuelle $X_i^k \Rightarrow X_i^{k+1}$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (2)$$

À quoi correspondent les paramètres w , c_1 , c_2

Compromis entre inertie et attractions

- w représente l'inertie de la particule ou *inertia weight*
- c_1 représente l'attraction vers sa meilleure position
 c_1 est également appelé le *cognitive coefficient*
- c_2 représente l'attraction vers la meilleure de ses voisines
 c_2 est également appelé le *social coefficient*

Choix des paramètres w , c_1 , c_2

- w est souvent une valeur proche de 1 ($0 < w \leq 1$)
- c_1 et c_2 sont souvent tirés dans $[0; \varphi_1]$ et $[0; \varphi_2]$, on peut aussi prendre $\varphi_1 = \varphi_2 = \varphi_{max}$ pour réduire les paramètres

$$\varphi_{max} = (w + 1)^2 / 2$$

“Grandes” lignes - N particules P_0 à P_{N-1} , P_i avec $i \in \{0, \dots, N-1\}$

On simplifie : la même meilleure position globale X^g pour tous les P_i

A. Initialisation

- 1 Initialiser les paramètres w, φ_1, φ_2
- 2 Initialiser le compteur d'itération $k = 0$
- 3 Initialiser aléatoirement la position X_i^0 des particules P_i
- 4 Initialiser aléatoirement la vitesse V_i^0 des particules
- 5 Initialiser ...

B. Optimisation

- 1 Mettre à jour les vitesses et positions des particules
- 2 Calculer $f_i^{k+1} = f(X_i^{k+1})$ pour toutes les particules X_i^{k+1} avec f qui est la fonction de coût (ou fonction objectif)
- 3 Si $f_i^{k+1} < f_i^b$ alors $f_i^b = f_i^{k+1}$ et $X_i^b = X_i^{k+1}$
- 4 Si $f_i^{k+1} < f^g$ alors $f^g = f_i^{k+1}$ et $X^g = X_i^{k+1}$
- 5 Incrémenter le compteur d'itération
- 6 Si critère d'arrêt vérifié allez en **C.** sinon reprendre en **B.1**

C. Terminaison $\Rightarrow X^g$ est l'optimum / la position cherchée

Essais particuliers - algorithme - 2/2

Description pseudo-code avec N , w , φ_1 , φ_2 comme paramètres

```
1:  $k = 0$ 
2:  $X_i^k$  = positions aléa des  $N$  particules;  $V_i^k$  = vitesses aléa. des  $N$  particules
3: Initialiser  $X_i^b, f_i^b, X^g$  et  $f^g$ 
4: repeat
5:    $i = 0$ 
6:   repeat
7:     Tirer aléatoirement  $c_1$  et  $c_2$ 
8:      $V_i^{k+1} = w \cdot V_i^k + c_1 \cdot (X_i^b - X_i^k) + c_2 \cdot (X^g - X_i^k)$ 
9:      $X_i^{k+1} = X_i^k + V_i^{k+1}$ 
10:     $f_i^{k+1} = f(X_i^{k+1})$ 
11:    if  $f_i^{k+1} < f_i^b$  then
12:       $f_i^b = f_i^{k+1}$ ;  $X_i^b = X_i^{k+1}$ 
13:      if  $f_i^{k+1} < f^g$  then
14:         $f^g = f_i^{k+1}$ ;  $X^g = X_i^{k+1}$ 
15:      end if
16:    end if
17:     $i = i + 1$ 
18:  until  $i = N$ 
19:   $k = k + 1$ 
20: until critère d'arrêt vérifié
```

Travail à faire - implanter l'algo. des essais particulières

Écrire du code implémentant les étapes suivantes

- 1 Écrire une fonction calculant $f(x_1, x_2) = F2(x_1, x_2)$ (cf. Cours 1)
- 2 Afficher $f(x_1, x_2)$ où $x_1, x_2 \in [-5; 5]$ avec pyplot
- 3 Écrire une fonction PSO qui prend en entrée :
 - la fonction de coût, le nombre de particules N ;
 - la valeur des paramètres w, φ_1, φ_2 ;
 - la dimension de la fonction de coût ;
 - le nombre maximum d'itérations ;et retourne la meilleure solution globale trouvée
- 4 Faire une animation (vidéo) montrant les particules trouvées au cours des itérations

Étudier la bibliothèque PySwarms

- Utiliser la bibliothèque pour trouver l'optimum de $F2$
- Tester le *Training a Neural Network* des Use cases