

swarm intelligence.

## Ant Colony Algorithm:

### Algorithm global:

- forecastea este înlocuită cu întreg textul.

### Notatii:

$w_i$  ~~not~~ cuvânt ambiguu.

$m$  = nr. cuv. din tot textul.

$w_{ij}$  = cel mai adecvat sens al lui  $w_i$ , fiind dat contextul (tot textul).

~~cel mai adeq~~  $d(w_i, j)$  = definiția unui sens  $j$  al cuv.  $w_i$

Spațiul de căutare corespunde tuturor combinațiilor posibile ale tuturor sensurilor tuturor cuv. din textul care se procesează.

(complexitate exponențială)

↓  
e nevoie de o căutare inteligentă.

$C$  = o configurație a cuvintelor.

reprezentarea unei configurații: un vector de întregi a.s.

$(j = C[i])$  = sensul  $j$  pt. cuv.  $w_i$

folonii furnici artificiale.

### Construirea grafului:

- fiecare no posibil al unui cur. este asociat unui nod curb, care va fi nod punto.

↳ curburile produc furnici care se mișcă în graf.

- furnicile produc mișcări reprezentat printr-o matrice care conține limita curbile din ~~smat~~ def. rezultatului coresp. curbului.

- Tipuri de noduri: → curb mamă (de unde a născut furnica)  
(din perspect.  
unei furnici)

→ curb dispan  $\phi$ : corespunde unui alt sens al aceluși curant.

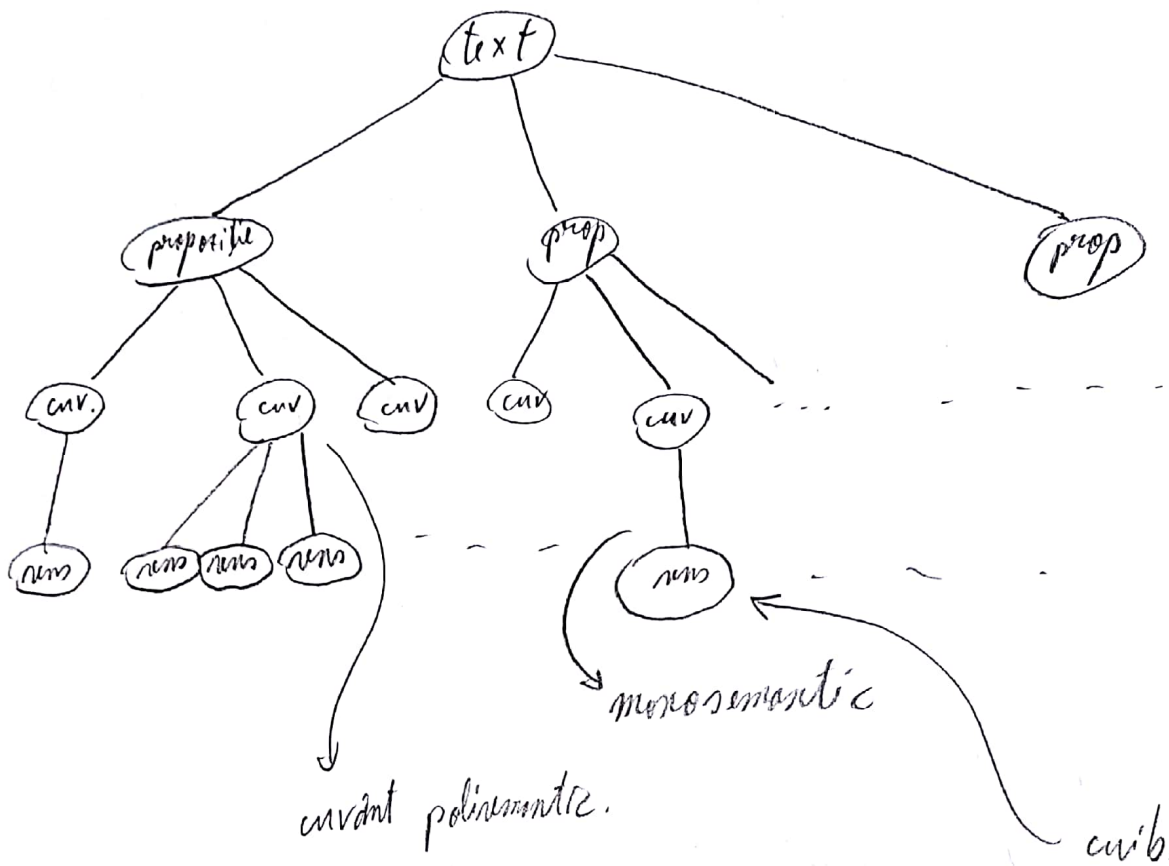
→ ~~curb potențial~~

→ potențial curb potențial: corespunde altor curvile.

→ nod simplu: orice nod care nu e nod.

↳ fiecare nod simplu îi este asociat un vector de mișcare (de lungime fixată; inițial vid).

representare sub formă de graf arbore:



persoanele sunt criaturi

Miscare furnici:

Depinde de:

- resursele atribuite de algoritmului local.
- prezența energiei în noduri
- trecerea prin noduri a altor furnici
- vectorii de mișcare ai nodurilor.

Reguli:

Când o furnică sosește în cuibul altui sens, ea poate fie să continue explorarea lui (în fct. de acor) sau poate decide să construiască un pod între acest cuib și cuibul mamei și să îl folosească pentru a se întoarce în cuibul mamei.

Pap

1.

2

3

Podurile se comportă ca orice altă mucle, cu excepția momentelor în care concentrația lor de peroxon devine 0, caz în care podul se morpă.

Cu cât sunt mai strâns legate ramurile a două ciiburi, cu atât mai mult podurile dintre aceste noduri vor contribui la partajarea resurselor dintre ele. Aceste ciiburi vor forma împreună meta-ciiburi. Se formează astfel un drum de interpretare.

Not

Un ciclu se compune din:

- se elimină furnicile moarte și podurile care nu mai au persoane
- compensator fiecarei ciib, se produce o nouă furnică.
- pentru fiecare furnică: se determină nivelul prin care se află (câtă energie nu se întoarce la ciibul mame)
- se mișcă
- se creează un pod de interpretare, dacă este posibil
- se actualizează nivelul (nivelul de energie al fiecarei nod, persoana și vectorii de mișcare.)

Pasi:

1. Studiem nașterea furnicilor, morțile, nivelul de energie.
2. Studiem mișcările furnicilor
3. Crearea și pierderea nodurilor; tipuri de produse
4. Modelul de feromoni care este creat
5. Mirosul

~~1. Studiem nașterea furnicilor, morțile, nivelul de energie.~~

Notatii:  $F_A$  = ~~cuiburile care conțin~~ cuibul compunător cuibului A

$f_A$  = furnica născută în cuibul  $F_A$

$v(x)$  = vector de miros asociat lui  $x$  ( $x$  poate fi furnică sau nod).

$E(x)$  = energie corp. lui  $x$  ( $x$  poate fi furnică sau nod).

$\#Eval_f(N)$  = evaluarea nodului  $N$ , făcută de furnica  $f$ .

$Eval_f(A)$  = eval. unei muchii  $A$  (cantitatea de feromoni corp. muchiei) făcută de furnica  $f$ .

$f(t/c)(A)$  = cantitatea de feromoni corp. muchiei  $A$  la momentul dat de  $t$  sau în cadrul ciclului dat  $c$ .

Parametri:

$E_{\alpha}$  = energia primită de o furnică când ajunge la un nod. Valorile sunt între 1-30.

$E_{max}$  = cantitatea maximă de energie care poate fi transportată de o furnică. Valori între 1-60.



- $\delta$  = rata de răspuns a fenomenului între două cicluri. Valori 0,0-1,0  
 $E_0$  = cantitatea inițială de energie care există în fiecare nod. Valori 5-60.  
 $\omega$  = ~~etate~~ <sup>durata</sup> de viață al unei furnici (exprimată în cicluri). 1-30 cicluri.  
 $L_v$  = lungimea vectorului de mișcare. 20-200  
 $S_v$  = procentaj al componentelor vectorului de mișcare depozitate de o furnică atunci când roșește într-un nod. 0-100 %.  
 $C_{ac}$  = nr. de cicluri ale simulării. 1-500.

### 1. Nașterea furnicilor, moartea și nivelul de energie:

#### inițializarea algoritmului:

- se atribuie o energie  $E_0$  fiecărui nod al mediului
- fiecare nod creșt  $N$  are posibilitatea să producă o furnică  $A$ , folosind o unitate de energie, cu probabilitate  $p(N_A)$ .  
 Prin def.,  $p(N_A) = \frac{\max(E(N))}{\omega} + 0,5$ , unde  $E(N)$  este energia nodului  $N$ .
- când furnica moare, energia acumulată de ea în timpul vieții ei este eliberată în nodul în care a murit.

## 2. Mișcările furnicilor:

- când o furnică se află într-un nod, ea atribuie o probabilitate de tranziție muchiilor care conduc spre toate nodurile vecine. Probabilitatea de a trece printr-o muchie  $A$  și pt. a ajunge în nodul  $N_i$  este  $P(N_i, A_j)$

$$P(N_i, A_j) = \frac{\text{Eval}_f(N_i, A_j)}{\sum_{k=1, l=1}^{K=n, L=m} \text{Eval}(N_k, A_l)}, \text{ unde } \text{Eval}_f(N, A) = \text{Eval}_f(N) + \text{Eval}_f(A)$$

pt. de evaluare a unui nod  $N$  atunci când ajunge în el, venind de-a lungul muchiei  $A$ .

- o furnică nou născută începe să caute mâncare în felul următor: → este atrasă de nodurile care au cea mai mare cantitate de energie.  $\text{Eval}_f(N) = \frac{E(N)}{\sum_{i=0}^n E(N_i)}$

→ evită să meargă de-a lungul muchiilor cu mult fenomen.

$$\text{Eval}_f(A) = 1 - \varphi_+(A)$$

- furnica colectează atâta energie cît este posibil pînă când decide să o aducă înapoi acasă și intră în modul de întoarcere cu prob.  $\frac{E(f)}{E_{\max}}$

→ când decide să se întoarcă, furnica se deplasează  
 urmând statistic muchiile care au cel mai mult  
 fenomen.

$$\text{Eval}_f(A) = \psi_f(A).$$

$$\text{Eval}_f(N) = \frac{\text{ExtLork}(V(N), \overbrace{V(f_A)}^{\substack{\text{vec. de mixer} \\ \text{asociat furnicii} \\ \text{măsurate în} \\ \text{cubul } F(A)}})}{\sum_{i=1}^k \text{ExtLork}(V(N_i), V(f_A))}$$