

Optimisation par colonies d'abeilles



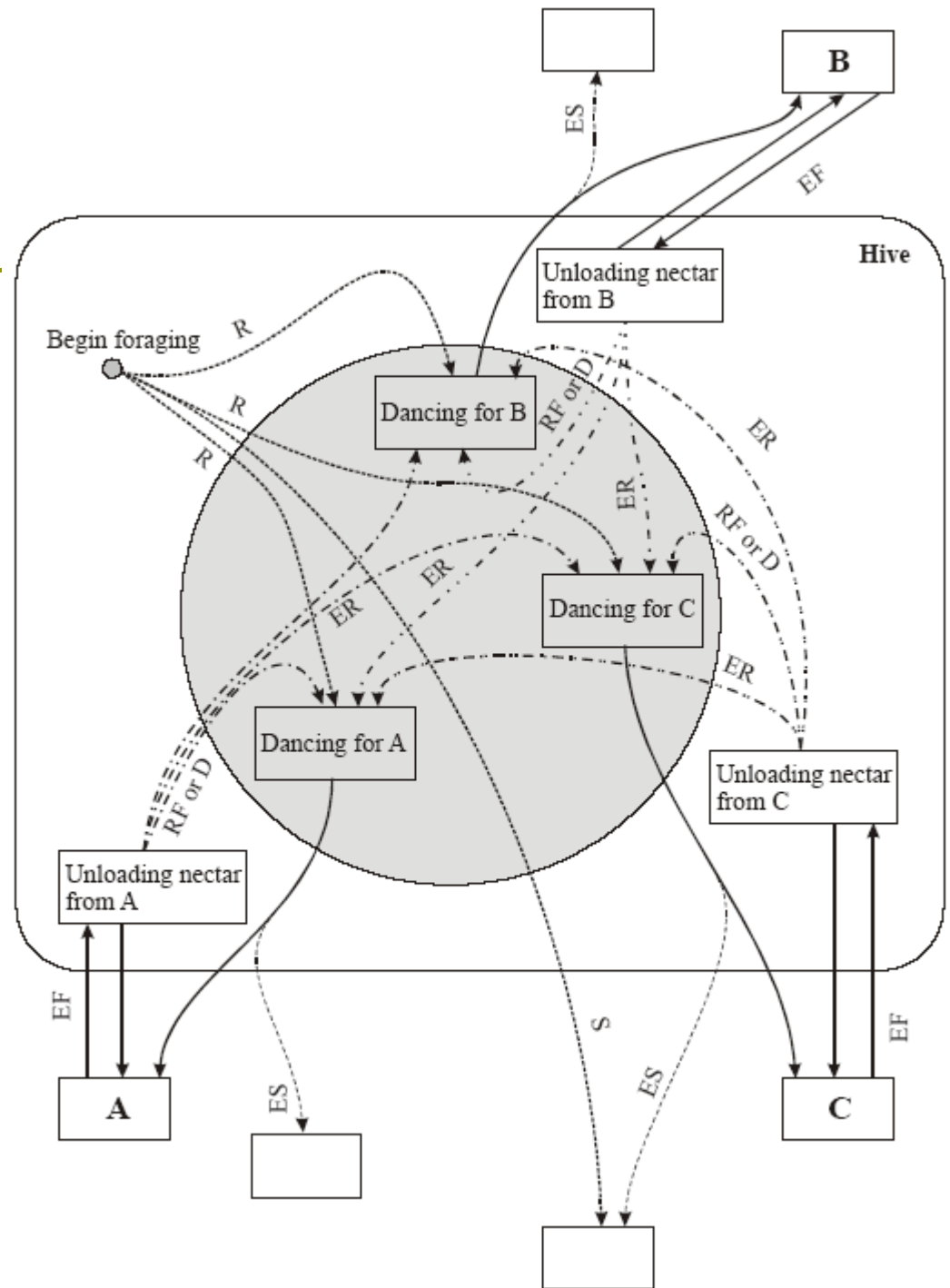
adapté de

Panta Lučić
Virginia Polytechnic Institute

Comportement des abeilles

- ❑ Les colonies d'abeilles sont en mesure de repérer des sources de nectar de bonne qualité.
- ❑ Le principe d'auto-organisation d'une colonie d'abeilles est basée sur un ensemble de règles simples qui gouverne le comportement individuel de chaque abeille.
- ❑ L'intelligence collective de la colonie émerge de l'échange d'informations entre les abeilles. Les échanges se font majoritairement en un lieu commun appelée la piste de danse.
- ❑ C'est en dansant qu'une abeille communique aux autres la localisation et la qualité d'une source de nectar.

- R = *Recruit*
- S = *Scout*
- F = *Forager*
- D = *Dancer*
- E = *Experienced*
- RF = *Reactivated forager*



Problème du voyageur de commerce

Le processus de recherche d'une solution se divise en:

- Itération: sélection aléatoire d'un sommet de départ qui correspond à la localisation de la ruche.
- Stades: Chaque itération est divisée en stades durant lesquels chaque abeille active visite s nœuds.

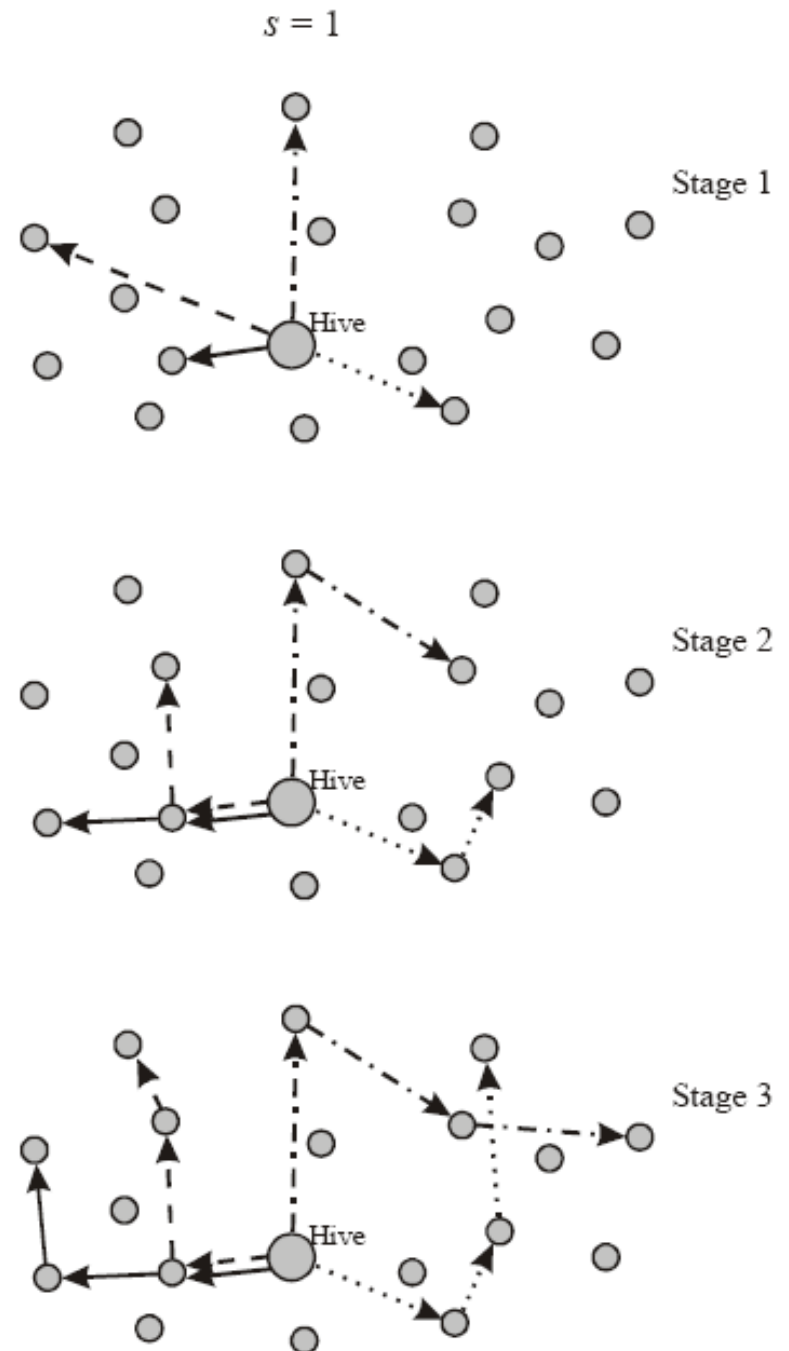
Processus de prise de décision

À la fin d'un stade, les abeilles doivent prendre une décision:

- Abandonner leur tour partiel et poursuivre avec le tour partiel d'une autre abeille.
- Poursuivre avec leur tour partiel sans recruter d'autres abeilles.
- Poursuivre avec leur tour partiel après avoir recruté d'autres abeilles.

Illustration

- Dans cet exemple, chaque abeille visite $s=1$ sommet à chacun des stades.
- Ceci est répété jusqu'à ce que des tours complets soient obtenus.



Notation

$B(u, z)$ = Nombre d'abeilles qui butinent durant le stade u ($u = 0, 1, 2, \dots, \left\lfloor \frac{|N|-1}{s} \right\rfloor$) de l'itération z .

$$b_k(u, z) = \begin{cases} 1, & \text{si la } k^{\text{ième}} \text{ abeille butine au stade } u \text{ de l'itération } z \\ 0, & \text{sinon} \end{cases}$$

Comme les abeilles ne commencent pas toutes à butiner en même temps, nous définissons la variable :

$$h_k(u, z) = \begin{cases} 1, & \text{si } w > r_k(u, z) \text{ et } b_k(u-1, z) = 0 \\ 0, & \text{sinon} \end{cases}$$

Où : w est un paramètre fourni par le programmeur.

$r_k(u, z)$ est un nombre aléatoire entre 0 et 1.

$g_k(u, z)$ = Dernier noeud visité par l'abeille k à la fin du stade u de l'itération z .

$N_k(u, z)$ = Ensemble des noeuds non-visités par l'abeille k au stade u de l'itération z .

$n_{ij}(z)$ = Nombre d'abeilles qui ont visité le lien (i, j) à l'itération z .

Sélection du prochain sommet à visiter

- Les abeilles choisissent le prochain sommet à visiter selon le modèle probabiliste suivant :

$$p_{ij}^k(u+1, z) = \begin{cases} \frac{e^{-ad_{ij}} \frac{z}{\sum_{r=\max(z-b, 1)}^{z-1} n_{ij}(r)}}{\sum_{l \in N_k(u, z)} e^{-ad_{il}} \frac{z}{\sum_{r=\max(z-b, 1)}^{z-1} n_{il}(r)}}, & i = g_k(u, z), j \in N_k(u, z), \forall k, u, z \\ 0, & \text{sinon} \end{cases}$$

Favorise les liens de courte distance

Ajuste l'influence de la distance

Nombre total d'abeilles ayant visité le lien (i,j) au cours des b itérations précédentes

Facteur de normalisation

Décision

- À la fin d'un stade, une abeille poursuit son tour partiel avec la probabilité suivante:

$$p_k(u+1, z) = e^{\frac{L_k(u, z) - \min_{r \in w(u, z)} (L_r(u, z))}{u z}}$$

- Ainsi, l'abeille qui a le tour partiel de longueur minimale le poursuit avec une probabilité de 1.

$$p_k(u+1, z) = e^{\frac{L_k(u, z) - L_k(u, z)}{u z}} = e^0 = 1$$

- Si l'abeille poursuit avec le même tour partiel, elle peut aussi décider de recruter de nouvelles abeilles avec la probabilité suivante:

$$p_{danse} = (1 - p^*) \text{ où } p^* \ll 1$$

Décision

- Si une abeille abandonne son tour partiel ou devient active, elle va sur la piste de danse afin de poursuivre le tour partiel d'une autre abeille.
- Les tours partiels ξ « annoncés » ont deux attributs:
 - Longueur totale
 - Nombre d'abeilles qui poursuivent ce tour partiel

$$\alpha_{\xi}(u, z) = \begin{cases} \frac{L_{\xi}(u, z) - \min_{r \in Y(u, z)} (L_r(u, z))}{\max_{r \in Y(u, z)} (L_r(u, z)) - \min_{r \in Y(u, z)} (L_r(u, z))}, & \text{pour } \max_{r \in Y(u, z)} (L_r(u, z)) \neq \min_{r \in Y(u, z)} (L_r(u, z)), \xi \in Y(u, z), \forall u, z \\ 0, & \text{sinon} \end{cases}$$

$$\beta_{\xi}(u, z) = \begin{cases} \frac{B_{\xi}(u, z) - \min_{r \in Y(u, z)} (B_r(u, z))}{\max_{r \in Y(u, z)} (B_r(u, z)) - \min_{r \in Y(u, z)} (B_r(u, z))}, & \text{pour } \max_{r \in Y(u, z)} (B_r(u, z)) \neq \min_{r \in Y(u, z)} (B_r(u, z)), \xi \in Y(u, z), \forall u, z \\ 0, & \text{sinon} \end{cases}$$

Sélection du tour partiel

- Une abeille choisit un nouveau tour partiel selon une probabilité qui dépend de la longueur du tour et du nombre d'abeilles qui l'ont exploré.

$$p_{\xi}(u, z) = \frac{e^{\rho\beta_{\xi}(u, z) - \theta\alpha_{\xi}(u, z)}}{\sum_{\tau \in Y(u, z)} e^{\rho\beta_{\tau}(u, z) - \theta\alpha_{\tau}(u, z)}} \quad \xi \in Y(u, z), \forall u, z$$

Où : ρ, θ sont des paramètres fournis par l'utilisateur

Procédure d'amélioration

- À la fin de chaque itération, une méthode de descente avec voisinage de type k -opt est appliquée.
- Trois voisinages ont été testés :
 - 2-opt
 - 3-opt
 - 3-opt (version courte)
- Dans l'article, l'auteur se limite à 100 itérations.

Résultats expérimentaux (2-opt)

Table 3.1 – The results obtained by the Bee System enriched with 2-opt heuristic

Problem	Number of nodes	Optimal Value (O)	The best value obtained by the Bee System (B)	$\frac{(B - O)}{O}$ (%)	Time required to find the best solution (seconds)	Average value obtained by the Bee System over 20 runs (A)	St. Dev. (SD)	$\frac{(A - O)}{O}$ (%)
Eil51	51	428.87	431.121	0.53%	44	433.758	1.37	1.14%
Berlin52	52	7544.366	7544.366	0%	18	7634.37	78.2	1.19%
St70	70	677.11	678.621	0.22%	238	684.275	3.53	1.06%
Pr76	76	108159	108790	0.58%	127	109444.6	461	1.19%
Kroal00	100	21285.4	21441.5	0.73%	58	21575.7	138.83	1.36%
Eil101	101	640.21	642.45	0.35%	146	665.62	7.94	3.97%
Tsp225	225	3859	4065.56	5.35%	2076	4113.71	27.3	6.6%
A280	280	2586.77	2740.63	5.95%	1855	2784.81	19.56	7.66%

Résultats expérimentaux (3-opt)

Table 3.2 – The results obtained by the Bee System enriched with 3-opt heuristic

Problem	Number of nodes	Optimal Value (O)	The best value obtained by the Bee System (B)	$\frac{(B - O)}{O}$ (%)	Time required to find the best solution (seconds)	Average value obtained by the Bee System over 20 runs (A)	St. Dev. (SD)	$\frac{(A - O)}{O}$ (%)
Eil51	51	428.87	428.87	0	37	428.87	0	0
Berlin52	52	7544.366	7544.366	0	1	7544.366	0	0
St70	70	677.11	677.11	0	22	677.11	0	0
Pr76	76	108159	108159	0	11	108159	0	0
Kroa100	100	21285.4	21285.4	0	10	21285.4	0	0
Eil101	101	640.21	640.21	0	1741	643.05	1.7	0.44%
Tsp225	225	3859	3876.05	0.44%	5153	3905.32	18.9	1.2%
A280	280	2586.77	2600.34	0.53%	13465	2627.45	12.31	1.57%

Résultats expérimentaux (3-opt version courte)

Table 3.3 – The results obtained by the Bee System enriched with 3-opt “short version” heuristic

Problem	Number of nodes	Optimal Value (O)	The best value obtained by the Bee System (B)	$\frac{(B - O)}{O}$ (%)	Time required to find the best solution (seconds)	Average value obtained by the Bee System over 20 runs (A)	St. Dev. (SD)	$\frac{(A - O)}{O}$ (%)
Eil51	51	428.87	428.87	0	29	428.87	0	0
Berlin52	52	7544.366	7544.366	0	0	7544.366	0	0
St70	70	677.11	677.11	0	7	677.11	0	0
Pr76	76	108159	108159	0	2	108159	0	0
Kroa100	100	21285.4	21285.4	0	10	21285.4	0	0
Eil101	101	640.21	640.21	0	61	643.07	1.84	0.45%
Tsp225	225	3859	3899.9	1.06%	11651	3909.69	9.19	1.31%
A280	280	2586.77	2608.33	0.83%	6270	2632.42	14.89	1.76%
Pcb442	442	50783.55	51366.04	1.15%	4384	51756.89	195.3	1.92%
Pr1002	1002	259066.6	267340.7	3.19%	28101	268965.6	1182.22	3.82%

Conclusion

- ❑ Le système de colonies d'abeilles couplé à une méthode de descente de type k -opt produit des solutions de bonne qualité.
- ❑ Toutefois, on ne mentionne pas le nombre d'abeilles utilisées pour résoudre chacun des problèmes, ni les valeurs des paramètres $a, b, w, \theta, \rho...$
- ❑ L'auteur mentionne que les solutions ont été trouvées des temps raisonnables.
 - Mais un temps de 1h 44min pour un problème de 280 sommets, c'est quand même assez long...