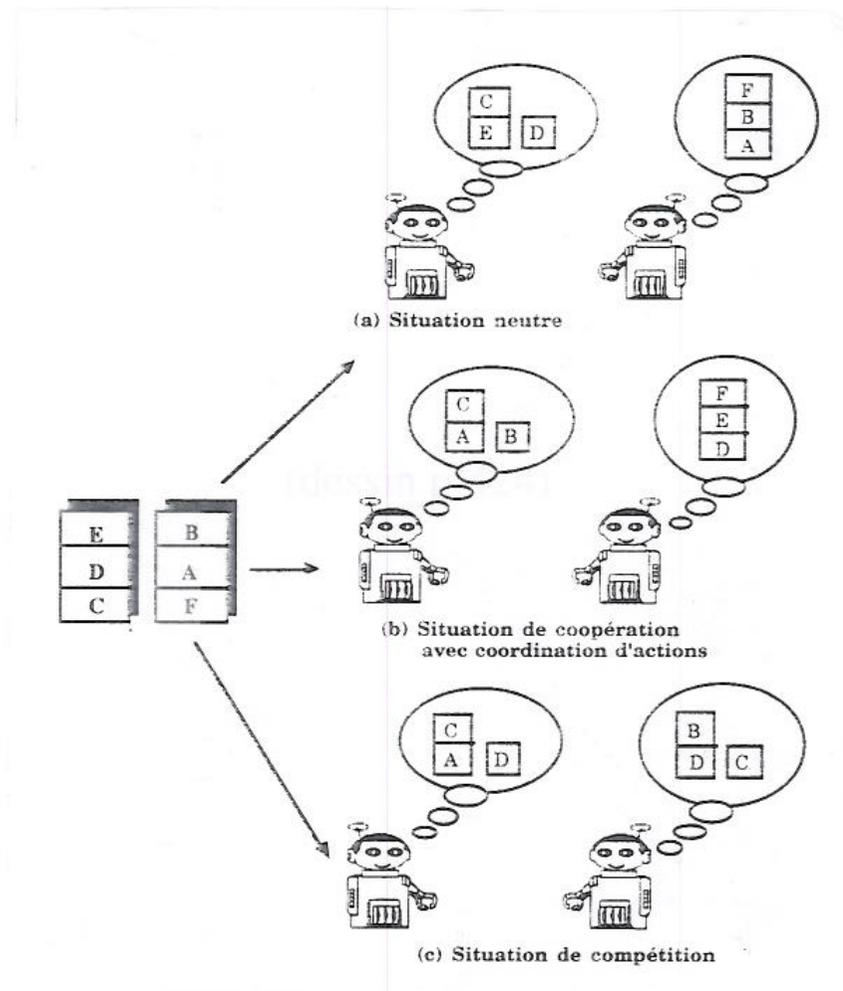


# 4. Systèmes Multi-Agent

## Coordination d'actions (Ferber 1995)





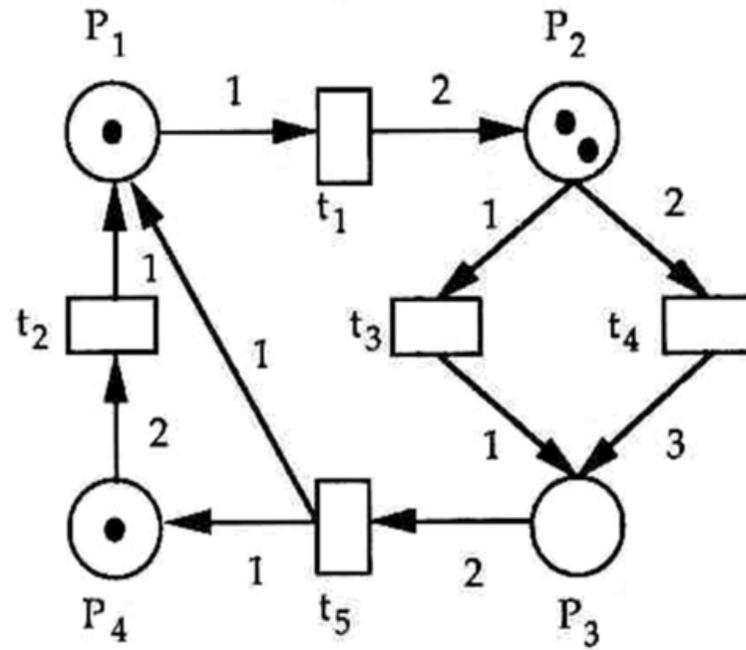
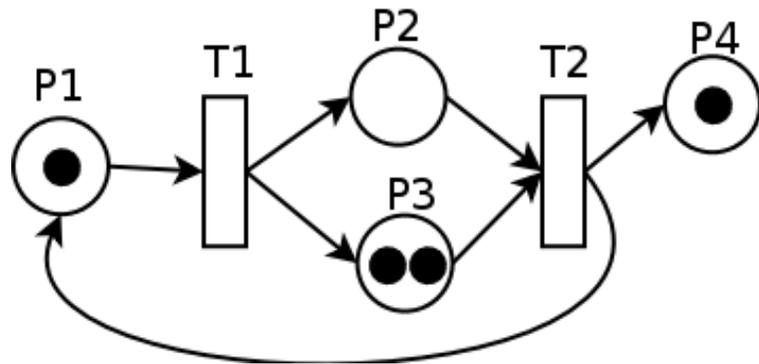
# Forme de coordination

- coordination par synchronisation
  - gérer la simultanéité de plusieurs actions
  - vérifier la cohérence des résultats
- coordination par planification
  - plans entraînant des conflits d'objectifs ou d'accès aux ressources
  - coordonner les plans
- coordination réactive
  - liaison perception-action in situ (non « a priori » comme les plans)
  - situations où il est difficile d'anticiper ce qui va se passer
- coordination par éco-résolution
  - coordination limitée
  - protocole suivi par tous les agents indépendamment du problème

# Synchronisation d'actions

Les réseaux de Petri : 6-uplet  $(S, T, F, M_0, W, K)$  où

- $S$  définit une ou plusieurs *places*.
- $T$  définit une ou plusieurs *transitions*.
- $F$  définit un ou plusieurs *arcs* (flèches).  
Un arc ne peut connecter que des paires place-transition.
- $M_0 : S \rightarrow \mathbb{N}$ , appelé *marquage initial*, où, pour chaque place  $s \in S$ , il y a  $n \in \mathbb{N}$  jetons.
- $W : F \rightarrow \mathbb{N}^+$  appelé ensemble d'*arcs primaires*, assignant à chaque arc  $f \in F$  un entier positif  $n \in \mathbb{N}^+$  qui indique combien de jetons sont *consommés* depuis une place vers une transition ou combien de jetons sont produits par une transition et arrivent pour chaque place.
- $K : S \rightarrow \mathbb{N}^+$  appelé *limite de capacité*, faisant correspondre à chaque place le nombre maximum de jetons qui peuvent occuper une place.



# Synchronisation d'actions

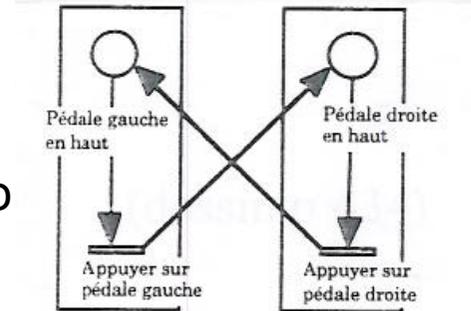
- Synchronisation de mouvements

Coordination de rythme et de positionnement des actions dans le temps en fonction des événements

exemple : coordination des jambes d'un cycliste

+ chaque jambe est considérée comme un agent

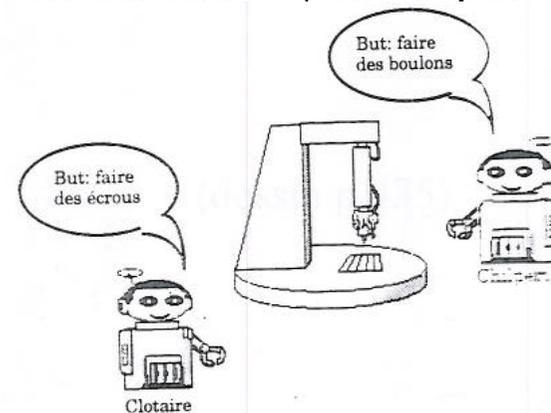
+ synchroniser la poussée de chaque jambe en fonction de la position des pédales



- Synchronisation d'accès à une ressource

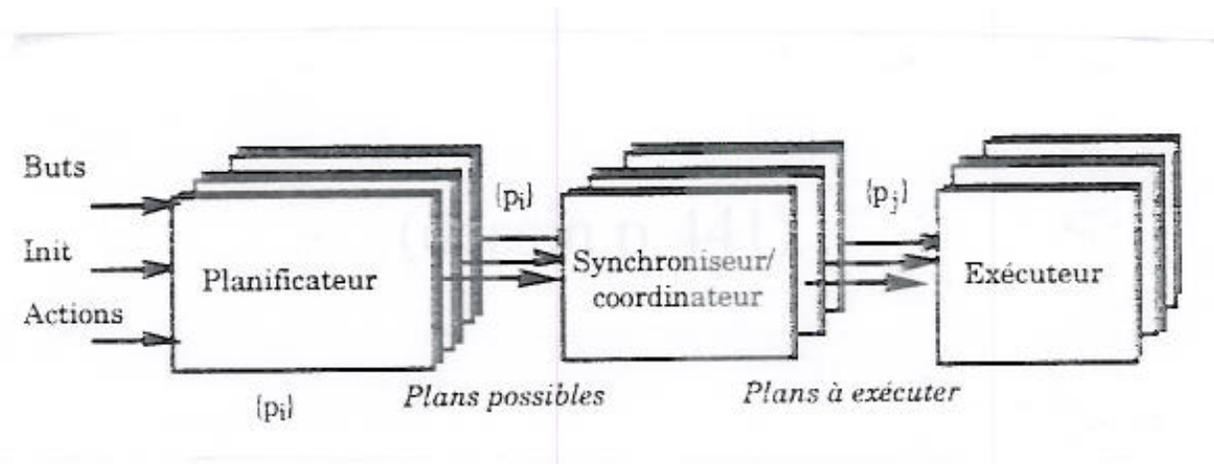
Section critique gérée par sémaphores

exemple : un robot demande à la machine son utilisation ; elle répond par un acquittement (bloque une autre utilisation à la fin, le robot envoie un message fin qui libère la machine



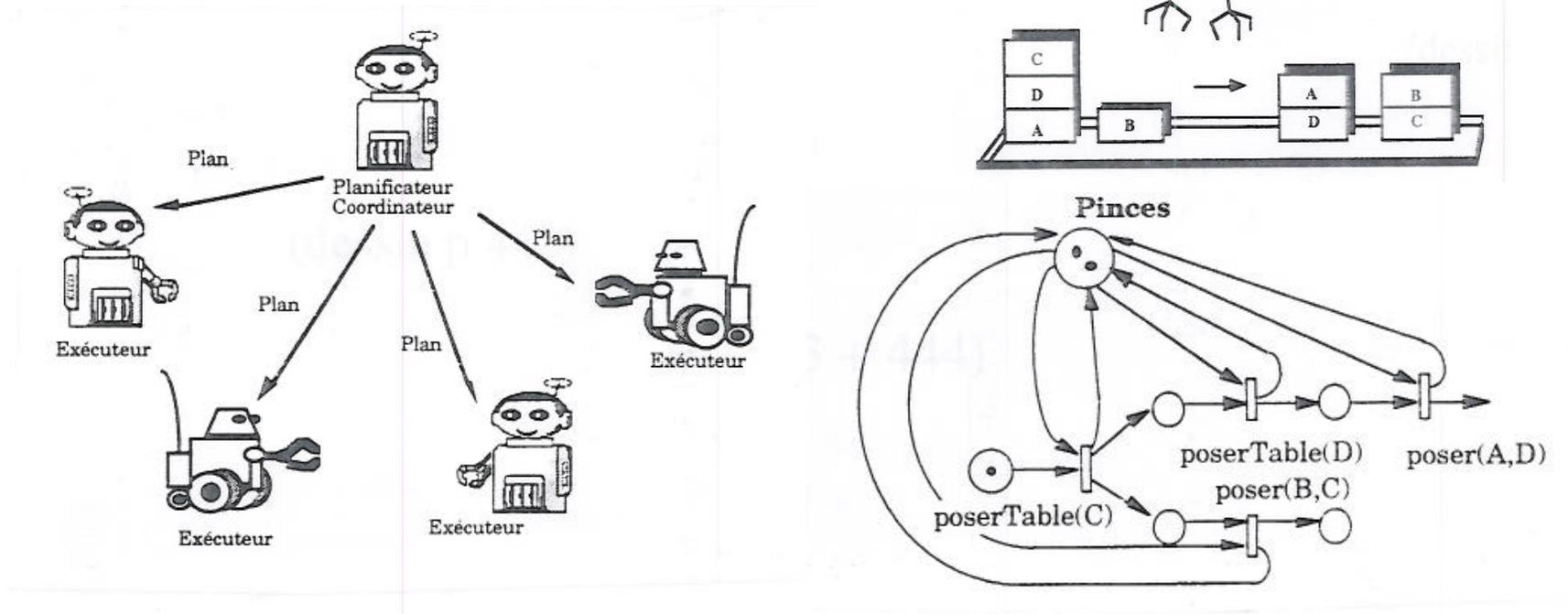
# Coordination par planification (1)

Utilise les résultats de la planification mono-agent et les étend aux situations multi-agents



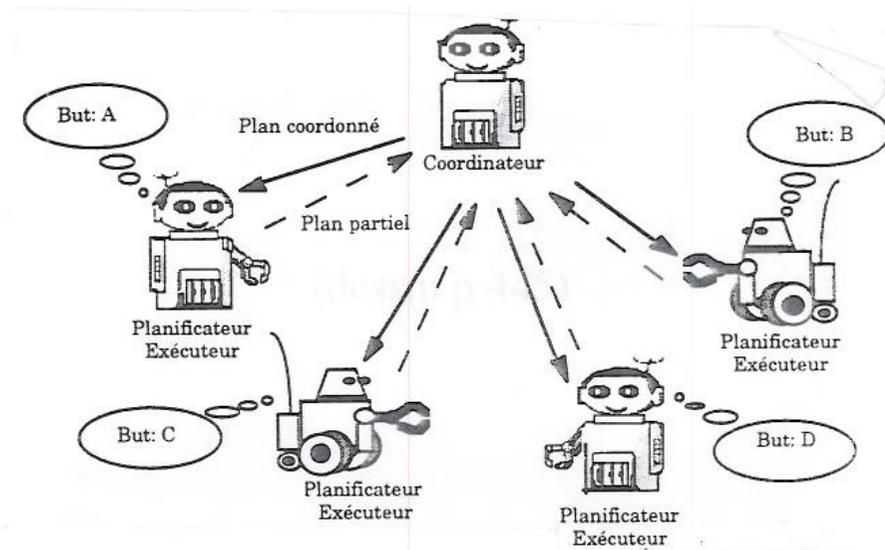
# Coordination par planification (2)

- Planification centralisée pour agents multiples  
un agent unique capable de construire des plans
  - + plan général partiel
  - + branches à exécuter en parallèle avec points de synchronisation
  - + allocation des tâches entre les agents



# Coordination par planification (3)

- Coordination centralisée pour plans partiels  
chaque agent construit son plan (partiel) ; un coordinateur central coordonne tous les plans partiels pour n'avoir qu'un seul plan global



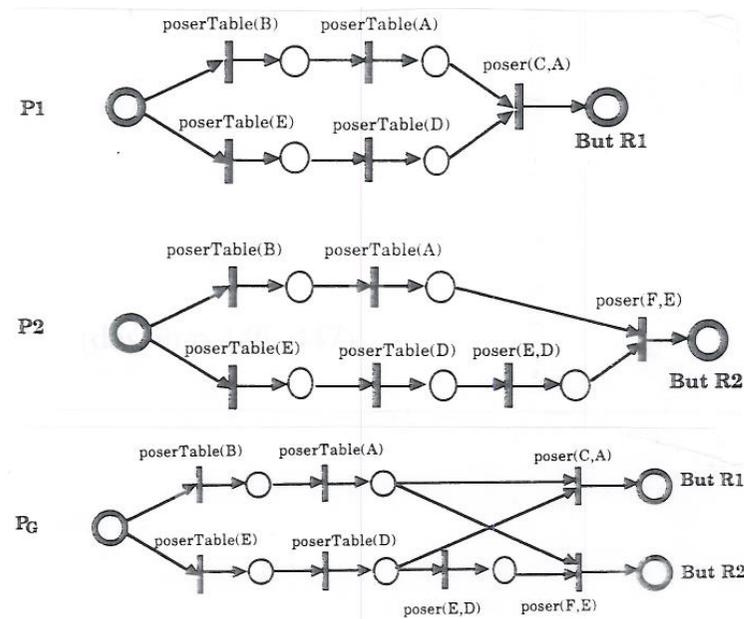
# Coordination par planification (4)

+ Coordination par fusion de plans (2 à 2)

→ identifier les places semblables

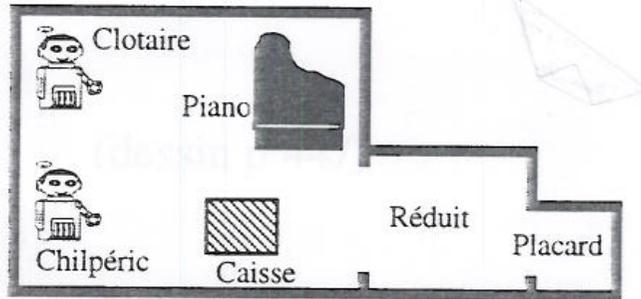
+ indépendances des actions : pas de fusion, les plans restent tels quels exécutés en //

+ relations positives entre actions : fusion possible, gérer les ressources en conflit, entrecroiser les actions des plans



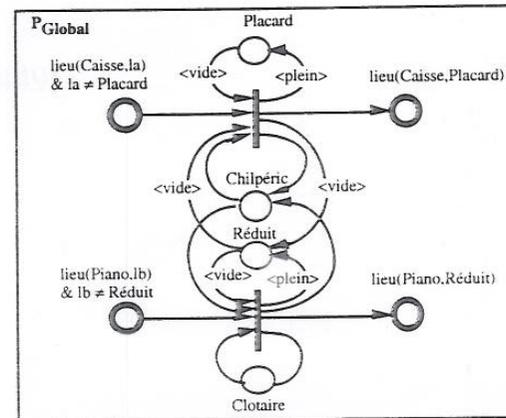
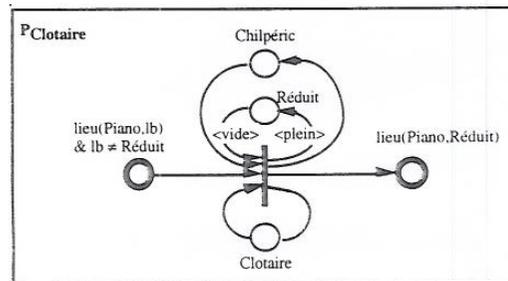
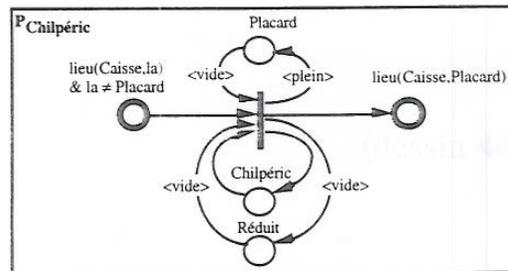
# Coordination par planification (5)

+ relations négatives et conflits de ressources : trouver une fusion de plan et déterminer un parcours dans l'exécution qui satisfasse tous les buts



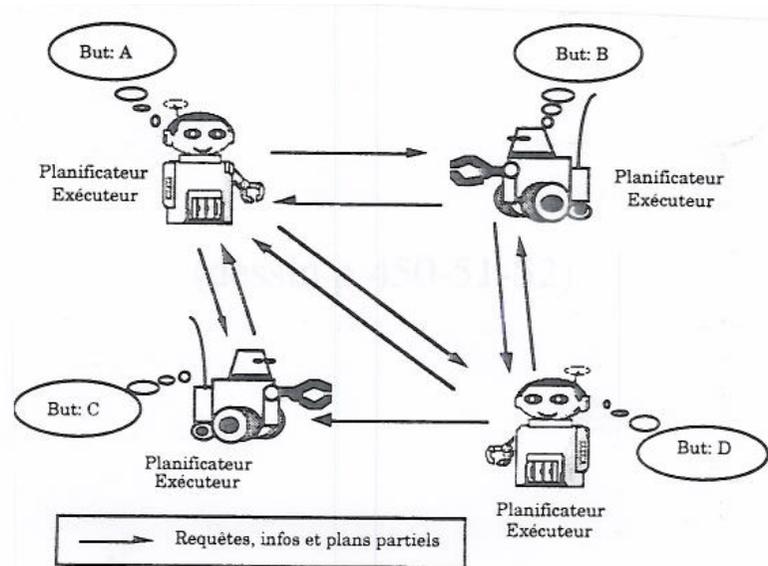
opérateur : `placerPiano`  
 nb agents : 2  
 pré : `lieu(Piano, x), vide(Réduit)`  
 retraits : `lieu(Piano, x), vide(Réduit)`  
 ajouts : `lieu(Piano, Réduit), plein(Réduit)`

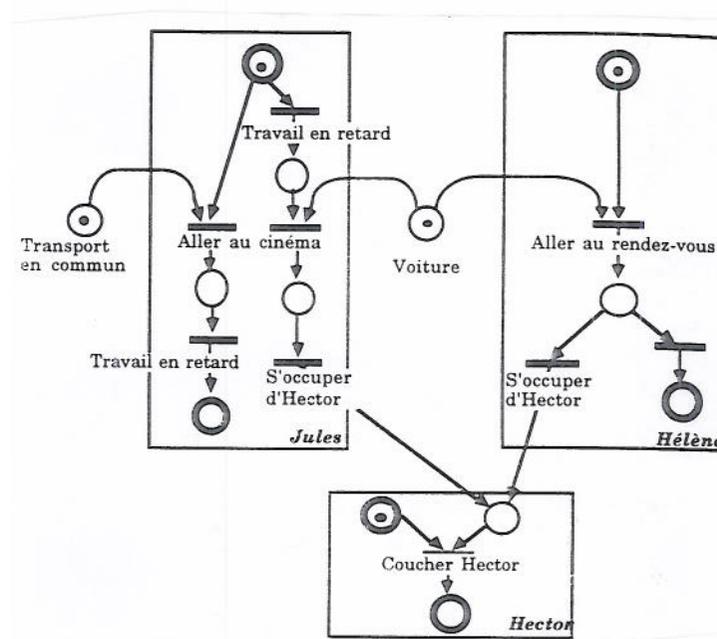
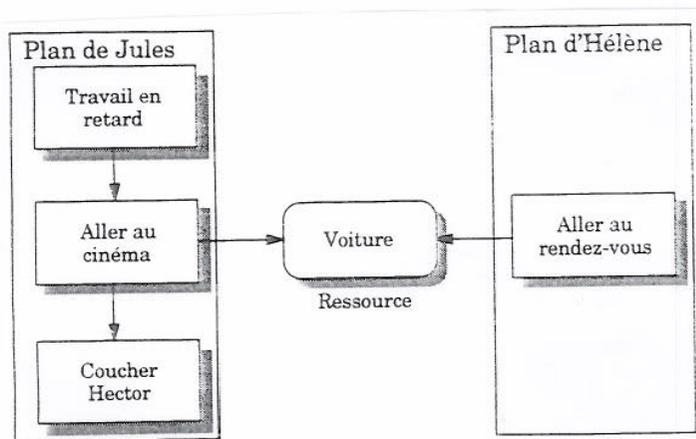
opérateur : `placerCaisse`  
 nb agents : 1  
 pré : `lieu(Caisse, y), vide(Réduit), vide(Placard)`  
 retraits : `lieu(Caisse, y), vide(Placard)`  
 ajouts : `lieu(Caisse, Placard), plein(Placard)`



# Coordination par planification (6)

- coordination distribuée pour plans partiels (planification distribuée)  
aucun agent centralisateur ni pour planifier, ni pour coordonner  
difficultés
  - résolution des conflits
  - reconnaissance des situations synergiques (services mutuels)





# Coordination réactive

- Marquer l'environnement pour coordonner les actions des agents
  - utilisation de champs de vecteurs pour la détermination des déplacements des agents mobiles
    - Meutes (division ou fusion possible)
      - maintenir une distance minimale par rapport aux autres agents
      - adapter sa vitesse à la moyenne de celle de ses voisins
      - aller vers le centre de gravité des agents voisins
    - utilisation de marques permettant d'utiliser l'environnement comme système de communication
      - Marques volatiles : fourmis
      - Marques non volatiles : araignées sociales

# Algorithmes de colonies de fourmis

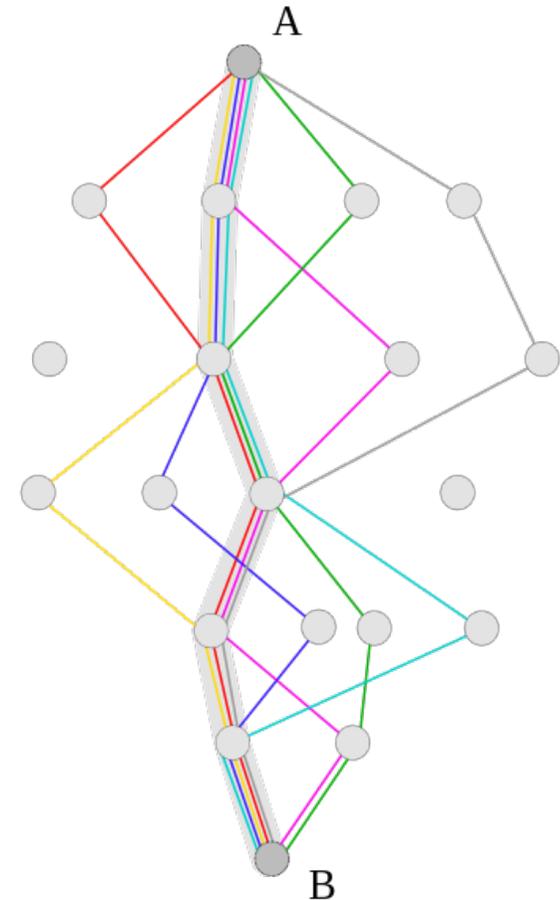


Un modèle de comportement est le suivant :

- une fourmi (appelée « éclaireuse ») parcourt plus ou moins au hasard l'environnement autour de la colonie ;
- si celle-ci découvre une source de nourriture, elle rentre plus ou moins directement au nid, en laissant sur son chemin une piste de phéromones ;
- ces phéromones étant attractives, les fourmis passant à proximité vont avoir tendance à suivre, de façon plus ou moins directe, cette piste ;
- en revenant au nid, ces mêmes fourmis vont *renforcer* la piste ;
- si deux pistes sont possibles pour atteindre la même source de nourriture, la plus courte sera, dans le même temps, parcourue par plus de fourmis que la plus longue ;
- la piste courte sera de plus en plus renforcée et donc de plus en plus attractive ;
- la longue piste finira par disparaître, les phéromones étant volatiles.

# Algorithmes de colonies de fourmis : applications

- Voyageur de commerce
- Avantage, par rapport aux autres metaheuristiques, dans le cas où le graphe étudié peut changer dynamiquement au cours de l'exécution  
→ La colonie de fourmis s'adaptera de façon relativement flexible aux changements.  
Intéressant pour le routage réseau.



# Araignées sociales

<http://webloria.loria.fr/~vthomas/recherche/araignees/>

Comportement d'une araignée virtuelle défini par 3 règles élémentaires :



- Dépot de soie :  
L'araignée virtuelle dépose des fils de soie dans son environnement en fonction d'une probabilité donnée.
- Attraction vers la soie :  
Les araignées virtuelles sont attirées par la soie et ont tendance à suivre un fil déjà déposé au sein de l'environnement en fonction d'une probabilité donnée et du nombre de fils de soie visibles par l'agent.
- Retour sur toile :  
Lorsqu'une araignée virtuelle se situe loin d'une toile, elle a une probabilité non nulle de suivre le fil qu'elle n'a pas attachée dans l'environnement pour retourner au dernier endroit où elle a déposé un fil.

# Araignées sociales : application

Extraire des régions contigües homogènes dans une image à niveau de gris.

Une toile correspond à une région extraite.

Une image peut être représentée par une matrice de pixels.

Dans cet environnement, nous disposons des agents dont le comportement est basé sur celui des araignées sociales.

Ces agents évoluent dans l'environnement et génèrent une toile censée représenter une région contigüe de niveau de gris homogène.

Le comportement d'un agent est décrit par les 3 règles suivantes :

- Se déplacer dans l'environnement sur une case adjacente ou en suivant un fil
- Poser un fil entre le piquet d'arrivée et le dernier piquet filé si le piquet d'arrivée correspond aux critères de sélection
- Revenir sur sa toile si l'agent s'en est trop éloigné

