

# Systèmes Multi-agents

## Planification

Claude Moulin

Université de Technologie de Compiègne

IA04

# Sommaire

- 1 Introduction
- 2 Action
- 3 Planification mono-agent
- 4 Planification multi-agent



# Questions faussement simples

- Qu'est-ce qu'un agent ?  
Comment décrire son comportement ?  
Comment décrire son architecture ?  
Comment décrire ses caractéristiques internes ?

# Théorie de l'action

- Conception classique de l'intelligence artificielle : la manifestation de l'intelligence repose sur un raisonnement logique.
- Systèmes multi-agents : étude du comportement des agents, de leurs actions sur l'environnement et des interactions qui en découlent.
- Théorie de l'action : permet de traiter les interactions et leurs conséquences sur l'environnement.

# Action : vision simplifiée

- Une action d'un agent vient modifier l'état de l'environnement.
- L'environnement passe d'un état stable à un autre état stable.

# Modèle d'actions

- Transformation d'un état global.
- Réponse à des influences.
- Processus informatique.
- Modification locale.
- Déplacement physique.
- Commande

Introduction

**Action**

Planification mono-agent

Planification multi-agent

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

Action : modification locale

Action : commande

# Sommaire

1 Introduction

2 **Action**

3 Planification mono-agent

4 Planification multi-agent



Introduction

**Action**

Planification mono-agent

Planification multi-agent

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

Action : modification locale

Action : commande

# Sommaire

1 Introduction

2 **Action**

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

Action : modification locale

Action : commande

3 Planification mono-agent

Planification classique

Planification avec contraintes temporelles sur des ressources

4 Planification multi-agent

# Transformation d'un état global

- Le monde possède un ensemble fini d'états possibles.
- Une action  $op$  est une transition d'un état  $e_1$  à un état  $e_2$   
 $e_2 = Executer(agent.op(< parametres >), e_1)$
- ex :  $e_2 = Executer(agent.mettreSur(B, A), e_1)$
- Condition : l'action est exécutable.

# Etat du monde

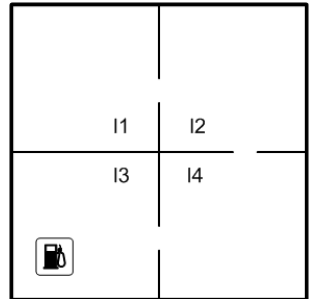
- Un état du monde est décrit comme un ensemble de formules atomiques dont la conjonction est supposée affirmer la validité de cet état.
- Les formule portent sur des instances de Concepts.

## Exemple de monde

- Structure :
  - Concepts : Lieu, Outil
  - Prédicats : nord, sud, est, ouest (Lieux communicants), lieuOutil
- Etat :
  - Individus : l1, l2, l3, l4 (Lieu) ; pompe (Outil)
  - lieuOutil(pompe,l3), ouest(l1,l2), ouest(l3,l4), nord(l2,l4), est(l2,l1), est(l4,l3), sud(l4,l2)
- Un état d'un monde est l'ensemble des assertions vraies à un instant donné.

## Exemple d'état du monde : RDF

```
:l1 a ont:Lieu .  
:l2 a ont:Lieu .  
:l3 a ont:Lieu .  
:l4 a ont:Lieu .  
:pompe a ont:Outil .  
:pompe ont:lieuOutil :l3 .  
:l1 ont:a_est :l2 .  
:l3 ont:a_est :l4 .  
:l2 ont:a_sud :l4 .  
:l2 ont:a_ouest :l1 .  
:l4 ont:a_ouest :l3 .  
:l4 ont:a_nord :l2
```



# Représentation des actions

- Les actions sont caractérisées par des opérateurs permettant de passer d'un état  $s_1$  à un état suivant  $s_2$ .
- Un opérateur a la forme d'un triplet :  
<pré-condition, suppressions, ajouts>
  - <pré-condition> : condition qui doit être vérifiée dans  $s_1$  pour que l'on puisse appliquer l'opérateur.
  - <suppressions> : ensemble de propositions décrivant les faits à supprimer de  $s_1$
  - <ajouts> : ensemble de propositions décrivant les faits à ajouter à  $s_1$  pour établir  $s_2$ .

## Exemple d'action - 1

- On considère un robot capable de se déplacer dans l'espace.
- Compléter la structure du monde et les états du monde par les assertions nécessaires.

```
:bot a ont:Robot .
```

```
:bot ont:positionRobot :12 .
```

## Exemple d'action - 2

- Opérateur allerSud :

```
Opérateur allerSud(?r)
```

```
  pré-condition : ?r ont:positionRobot ?x .  
                ?x ont:a_sud ?y .
```

```
  suppressions : ?r ont:positionRobot ?x .
```

```
  ajouts : ?r ont:positionRobot ?y .
```

```
fin
```



## Application de l'opérateur

- L'application de l'opérateur à la situation initiale  $s_1$  produit un nouvel état  $s_2$  dans lequel le robot s'est déplacé du lieu  $l_2$  vers  $l_4$ .

```
s1 = { :bot ont:positionRobot :l2 .  
      :l2 ont:a_sud :l4 . ... }  
s2 = Exécuter(allerSud(:bot), s1)  
s2 = { :bot ont:positionRobot :l4 .  
      :l2 ont:a_sud :l4 . ... }
```

- Les opérateurs ne décrivent que ce qui se transforme. Tout ce qui n'apparaît pas dans l'opérateur reste inchangé d'un état à l'autre.

# Postulats de ce modèle

- Le monde est **statique**.  
Les seules évolutions possibles du monde sont le fruit des actions des agents.
- Les actions sont **séquentielles**.  
Deux actions ne peuvent se produire en même temps.
- Les actions sont décrites par leurs **résultats**.  
Il faut décrire aussi les effets des résultats.

# Transformation d'un état global : problème

- Confusion entre ce qui est produit par les agents et ce qui se produit effectivement.
- Confusion entre le geste et le résultat du geste.
- Un agent exécute un geste mais le résultat du geste dépend des gestes des autres agents.
  - Un geste devrait être considéré comme une influence et le résultat de plusieurs gestes est obtenu en appliquant les lois de l'univers.
  - L'environnement compose l'ensemble des influences et en déduit le nouvel état du monde.

Introduction

**Action**

Planification mono-agent

Planification multi-agent

Action : transformation d'un état global

**Action : réponse à une influence**

Action : processus informatique

Action : déplacement physique

Action : modification locale

Action : commande

# Sommaire

1 Introduction

**2 Action**

Action : transformation d'un état global

**Action : réponse à une influence**

Action : processus informatique

Action : déplacement physique

Action : modification locale

Action : commande

3 Planification mono-agent

Planification classique

Planification avec contraintes temporelles sur des ressources

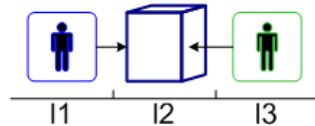
4 Planification multi-agent

# Réponse à une influence

- Principes d'influences et de réactions aux influences.
- Ce modèle reprend certains des éléments de l'approche précédente et en étend le modèle.
- Elle prend en compte les conséquences des actions simultanées des agents (interactions entre agents).
- Le modèle permet de décrire les actions considérées comme des déplacements dans un espace physique (application de la mécanique classique).

# Exemple

```
:l1 a ont:Lieu .  
:l2 a ont:Lieu .  
:l3 a ont:Lieu .  
:cube a ont:Cube .  
:a a ont:Agent .  
:b a ont:Agent .  
:cube ont:position :l2 .  
:a ont:position :l1 .  
:b ont:position :l3 .
```



## Modèle : état, influence

- Un état du monde est décrit à l'aide d'un ensemble de formules atomiques de la forme  $p(a_1, \dots, a_n)$  où  $p$  est un prédicat d'arité  $n$  et les  $a_1, \dots, a_n$  sont des constantes ou des termes fonctionnels ne comportant pas de variables.
- Les influences représentent les gestes, ou tentatives d'actions, accomplis par les agents à partir de l'état courant du monde.  
Elles sont décrites aussi à l'aide d'un ensemble de formules atomiques.

## Modèle : actions

- Les actions sont caractérisées par des opérateurs produisant des influences.
- Un opérateur a la forme :  
<pré-condition, post-condition>
  - <pré-condition> : condition qui doit être vérifiée dans un état  $\sigma$  pour que l'on puisse appliquer l'opérateur.
  - <post-condition> : ensemble des influences produites.



## Modèle : actions

- Les actions sont caractérisées par des opérateurs produisant des influences.
- Un opérateur a la forme :  
<pré-condition, post-condition>
  - <pré-condition>, <post-condition> sont des ensembles de formules de la forme  $p(a_1, \dots, a_n)$  où  $p$  est un prédicat d'arité  $n$  et les  $a_i$  sont des constantes ou des variables.
  - Exécuter( $op, \sigma$ ) =  $\gamma$   
 $\gamma$  = <post-condition> si <pré-condition> vérifié dans  $\sigma$   
et {} sinon

## Exemple d'action - 2

- Opérateur déplacer :

Opérateur déplacer(?ag,?cb,?d)

pré-condition : ?ag a ont:Agent .

?cb a ont:Cube .

post-condition : poussée(?cb,?d) .

fin

$\gamma = \text{Exécuter}(\text{déplacer}(:a, :cube, +1))$

$\gamma = \text{poussée}(:cube, +1)$

## Composition d'actions

- $\text{Exécuter}(\text{op}_1 \parallel \text{op}_2, \sigma) = \text{Exécuter}(\text{op}_1, \sigma) \cup \text{Exécuter}(\text{op}_2, \sigma)$
- Les lois de l'univers sont décrites à l'aide d'une fonction Réaction qui précise comment le monde se transforme d'un état à un autre, en réagissant aux influences.
- Dans un environnement où les cubes réagissent aux poussées en se déplaçant d'une distance proportionnelle :

Réaction =

état : ?cb ont:position ?x .

influences : poussée(?cb, ?d)

faire: supprimer{?cb ont:position ?x}

ajouter{?cb ont:position place(?x, dep) }

où : dep = somme(?d) tq poussée(?cb, ?d)

## Réponse à une influence : avantage

- Exemple 1 : mécanique classique

L'état du monde est donné par la position et la vitesse d'un ensemble de corps.

Les influences sont les forces qui s'exercent sur les corps à un instant  $t$ .

L'application des lois de la mécanique consiste à faire le bilan des forces en présence et à en déduire la position et la vitesse des corps à l'instant  $t + \delta t$ .

- Exemple 2 : contrôle

Les influences produisent des modifications des variables de commandes du système à contrôler. Les lois de l'univers correspondent à la réponse du système à ces valeurs d'entrées.

Introduction

**Action**

Planification mono-agent

Planification multi-agent

Action : transformation d'un état global

Action : réponse à une influence

**Action : processus informatique**

Action : déplacement physique

Action : modification locale

Action : commande

# Sommaire

1 Introduction

**2 Action**

Action : transformation d'un état global

Action : réponse à une influence

**Action : processus informatique**

Action : déplacement physique

Action : modification locale

Action : commande

3 Planification mono-agent

Planification classique

Planification avec contraintes temporelles sur des ressources

4 Planification multi-agent

# Processus

- On considère que le monde est composé d'un ensemble de processus, c'est-à-dire de programme en cours d'exécution.
- On s'intéresse à l'ensemble de ces entités, à leur comportement et aux interactions qui s'établissent entre elles.
- On peut représenter un processus sous forme :
  - automates à états finis
  - automates à registres (réseaux de transitions augmentés)
  - réseaux de Petri

Introduction

**Action**

Planification mono-agent

Planification multi-agent

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

**Action : déplacement physique**

Action : modification locale

Action : commande

# Sommaire

1 Introduction

**2 Action**

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

**Action : déplacement physique**

Action : modification locale

Action : commande

3 Planification mono-agent

Planification classique

Planification avec contraintes temporelles sur des ressources

4 Planification multi-agent

# Déplacement

- Cette approche s'avère particulièrement intéressante dans le cas d'agents réactifs qui doivent planifier leur trajectoire, éviter d'entrer en collision, être capables de se rencontrer ou d'évoluer ensemble.
- Exemples : vol simultanés de drones, flux de voitures sur une route principale, particules.
- Le comportement d'un agent se manifeste comme un mouvement dans un espace euclidien en fonction du temps.



# Calcul vectoriel

- Mobile : au temps  $t_0$  l'agent se trouve au point M, il se trouvera au point M' au temps  $t_0 + \Delta_t$  ayant accompli le déplacement D résultant de forces :  $\overrightarrow{OM'} = \overrightarrow{OM} + \overrightarrow{D}$
- Particules : on peut définir des champs de forces en chaque point de l'espace ainsi que leur résultante.

## Cas d'utilisation

- Les actions des agents se ramènent naturellement à des déplacements dans un espace métrique.  
En particulier, tous les systèmes multi-agents qui conçoivent les agents comme des sortes de petits ou minuscules (swarm agents) robots qui se déplacent dans un environnement peuvent en tirer facilement profit.
- Les actions doivent être décomposées dans le temps et il est possible d'utiliser des notions de champs pour décrire les tendances des agents à effectuer quelque chose.

Introduction

**Action**

Planification mono-agent

Planification multi-agent

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

**Action : modification locale**

Action : commande

# Sommaire

1 Introduction

2 **Action**

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

**Action : modification locale**

Action : commande

3 Planification mono-agent

Planification classique

Planification avec contraintes temporelles sur des ressources

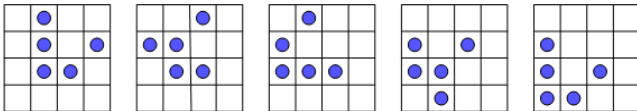
4 Planification multi-agent

## Aspect local

- L'action est vue comme une modification locale de l'environnement et ne concerne que les entités qui se trouvent proches de l'agent.
- L'action peut être vue comme une cause locale dont les effets se propagent de proche en proche dans les autres portions de l'espace.
- Le monde n'est plus un ensemble de propositions valides mais un réseau dont les nœuds représentent des entités réelles (portions d'espace, objets, êtres) ou bien abstraites (idées, situations) et les arcs correspondent aux liens qui existent entre ces entités (liens de contact, d'acointance, de dépendance).

## Exemple : automate cellulaire

- Le jeu de la vie :
  - Naissance : si l'automate (une cellule) est mort, il devient vivant si exactement trois de ses voisins sont vivants. Dans tous les autres cas, il reste mort.
  - Mort : si l'automate est vivant, il reste vivant si 2 ou 3 de ses voisins sont vivants. Il meurt dans les autres cas.



Introduction

**Action**

Planification mono-agent

Planification multi-agent

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

Action : modification locale

**Action : commande**

# Sommaire

1 Introduction

**2 Action**

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

Action : modification locale

**Action : commande**

3 Planification mono-agent

Planification classique

Planification avec contraintes temporelles sur des ressources

4 Planification multi-agent

# Aspect cybernétique - 1

- L'action consiste à faire varier un certain nombre de paramètres d'entrée (variables de commandes), d'un système physique pour obtenir des valeurs particulières des variables de sorties.
- Exemple : dans un problème de régulation, les variables de sortie doivent atteindre et garder des valeurs de consigne fixées.

## Aspect cybernétique - 2

- L'action n'est plus seulement une transformation d'états du monde, ni un simple mouvement, mais une activité complexe entièrement dirigée vers un but.
- Elle tient compte des réactions du milieu et des corrections à apporter aux actions précédentes.
- Cette conception de l'action comme commande est à la base de tous les systèmes de régulation et de contrôle de machines.



- 1 Introduction
- 2 Action
- 3 Planification mono-agent**
- 4 Planification multi-agent

# Sommaire

## 1 Introduction

## 2 Action

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

Action : modification locale

Action : commande

## 3 Planification mono-agent

Planification classique

Planification avec contraintes temporelles sur des ressources

## 4 Planification multi-agent

# Planifier avec des opérateurs de type STRIPS

- La modélisation de l'action comme transformation d'un état global est bien adaptée aux problèmes de planification classique.
- Une situation initiale  $\sigma_{init}$  et une situation finale  $\sigma_{fin}$  que l'on désire atteindre.
- Un ensemble d'opérateurs représentant les actions possibles d'un agent sur l'environnement.
- Résoudre un problème par planification c'est trouver une suite d'opérateurs  $X_1, \dots, X_i, \dots, X_n$  tel que :  
$$\sigma_{fin} = \text{Exécuter}(X_n \circ \dots \circ X_i \circ \dots \circ X_1, \sigma_{init})$$

# Planificateur

- Une recherche de plan est vu comme une résolution de problème.
- A partir de la spécification des situations initiales, des buts et de l'ensemble des actions disponibles, le planificateur trouve un ensemble de plans  $\{p_i\}$  parmi lesquels le plan à exécuter est choisi.

# Méthode par chaînage avant

- La méthode recherche un ensemble de faits en partant de l'état initial, faits qui peuvent s'avérer inutiles dans le problème à résoudre.
- Il a été montré que la recherche dans un espace d'états en chaînage avant n'est pas la méthode la plus efficace.
- Comme les problèmes de planning concernent de nombreux espaces d'états il est sans espoir de pouvoir résoudre un problème de cette manière sans une heuristique précise.

# Méthode par chaînage arrière

- La recherche part du but à atteindre et applique les actions en arrière jusqu'à trouver une séquence de pas qui atteigne l'état initial.
- On ne considère que des actions qui sont pertinentes (en lien avec le but).
- A chaque pas il y a un ensemble d'états à considérer.
- La recherche construit un arbre ET / OU.

# Représentation des actions

- Un opérateur d'action a la forme d'un triplet :  
<pré-condition, suppressions, ajouts>
  - <pré-condition> : condition à vérifier pour appliquer l'opérateur.
  - <suppressions> : ensemble de propositions décrivant les faits à supprimer de l'état courant.
  - <ajouts> : ensemble de propositions décrivant les faits à ajouter.
- Au dernier pas il suffit d'avoir un opérateur dont un des ajouts soit le but à atteindre ;  
il peut en exister plusieurs : Arbre OU.
- Au pas précédent les préconditions deviennent les nouveaux buts à atteindre : Arbre ET.

# Heuristique : relaxation

- On définit un problème plus simple et donc plus facile à atteindre en relâchant certaines conditions.
- On relâche les préconditions et les effets qui ne sont pas basées sur des littéraux (variable = valeur).
- On peut aussi sélectionner les préconditions à relâcher.
- On peut aussi ignorer la liste des suppressions ; la progression vers le but devient monotone.
- Ces types d'heuristiques ne raccourcissent pas le nombre d'actions successives à entreprendre.



# Heuristique : monde abstrait

- On définit un problème plus simple en travaillant dans un monde abstrait où le nombre de faits possibles est plus faible.
- Dans ce monde abstrait une solution est en général plus courte.
- Une solution abstraite est plus facile à étendre vers une solution réelle du problème original.

## Heuristique : conditions indépendantes

- On peut décomposer le problème à résoudre en plusieurs parties supposées indépendantes et composer ensuite les résultats.
- L'hypothèse de l'indépendance des sous-buts peut être optimiste ou pessimiste.
  - optimiste : interactions négatives entre les sous-plans (une action dans un plan supprime un but résolu par un autre plan).
  - pessimiste : des sous-plans contiennent des buts redondants (deux actions pourraient être remplacées par une seule dans le plan reconstitué).

## Autres approches

- Planification comme un problème de déduction logique (logique du premier ordre).
- Planification comme un système de résolution de contraintes (CHOCO : <http://choco.emn.fr/>).
- Planification comme raffinement de plans partiellement ordonnés.

# Sommaire

## 1 Introduction

## 2 Action

Action : transformation d'un état global

Action : réponse à une influence

Action : processus informatique

Action : déplacement physique

Action : modification locale

Action : commande

## 3 Planification mono-agent

Planification classique

Planification avec contraintes temporelles sur des ressources

## 4 Planification multi-agent

# Planifier d'abord, organiser ensuite

- Phase de planification dans laquelle les actions sont sélectionnées en respectant des contraintes d'ordre pour satisfaire le problème.
- Phase dans laquelle des informations temporelles sont ajoutées à un plan pour respecter les contraintes temporelles d'accès aux ressources.

# Type d'action

- Chaque action a une durée et un ensemble de contraintes sur des ressources.
- Chaque contrainte spécifie un type de ressources (outils, personnes), le nombre de ressources et la réutilisabilité de la ressource.
- Les ressources peuvent aussi être produites par des actions.
- Une solution doit indiquer la date de début de chaque action.
- Une solution peut être évaluée par une fonction de coût (une version simplifiée est la somme des durées des actions).

# Sommaire

- 1 Introduction
- 2 Action
- 3 Planification mono-agent
- 4 Planification multi-agent**

## Situations extrêmes

- Un agent seul observe l'environnement, planifie ses tâches et agit.
- Plusieurs agents agissent dans l'environnement. Chaque agent affronte un problème de planification multi-agents dans lequel il essaie de satisfaire ses propres buts avec l'aide des autres ou en opposition avec les autres (stratégies de jeu).



## Situation intermédiaire

- Planification multi-effecteur :
  - Un agent (assistant personnel) ayant plusieurs effecteurs (modalités) qui peuvent opérer de façon concurrente. Chaque modalité (parole, toucher, vision, etc.) doit avoir une planification particulière pour gérer les interactions positives et négatives.  
Multi-modalité parallèle.

# Planification centralisée

- Planification centralisée ; exécution découplée :
  - En fait la planification est centralisée mais la phase d'exécution est répartie entre les agents.  
Le sous-plan construit pour chaque agent peut nécessiter des actions de communication explicites avec d'autres entités.  
Quand la communication est soumise à certaines contraintes on a affaire à une planification décentralisée.

# Planification décentralisée

- Planification décentralisée :
  - Exemple : des robots de reconnaissance couvrant une large zone peuvent être dans des zones où la communication avec l'agent principal n'est pas possible et doivent partager ce qu'ils trouvent lorsque la connexion redevient possible.
- Certains systèmes offrent une situation mixte entre une planification centralisée et une planification décentralisée.

# Coordination

- Lorsqu'il n'y a qu'un seul agent qui fait la planification il n'y a qu'un but que chaque entité partage.
  - L'agent planificateur indique aux entités dépendantes les actions à exécuter.
- Quand des entités distinctes font leur propre planification elles peuvent partager le même but.
  - Quand plusieurs entités planifient, même avec le même but, elles peuvent décider de faire la même action en même temps, ce qui peut être anti-productif.  
Sans méthode de coordination, la planification multi-agent est souvent ingérable.

# Problèmes

- Représentation et planification d'actions multiples simultanées (multi-acteurs).
- Coopération, coordination, et compétition dans de véritables planifications multi-agents.

## Premier cas : synchronisation - 1

- Un plan correct est un plan qui exécuté par les acteurs, atteint le but.
- Une action est remplacée par une action multiple (nuplet d'actions de chaque agent) :  
 $\{a_1, a_2, \dots, a_n\}$
- Systèmes faiblement couplés :
  - on considère les actions indépendantes :  
A une itération donnée un agent peut ne pas agir ou agir d'une façon considérée indépendante des autres.

## Premier cas : synchronisation - 2

- Systèmes plus fortement couplés :
  - Chaque action en plus de sa description contient une liste d'action concurrentes.  
Le planificateur connaît ainsi les actions qu'il ne doit pas confier simultanément.

## Second cas - 1

- Chaque agent construit son propre plan (le but et les connaissances sont partagées).
- Chaque plan considère les actions des autres agents.
- Les agents doivent coopérer et lorsque plusieurs plans sont possibles, ils doivent être certains que les autres vont choisir ce plan.



## Second cas - 2

- Convention :
  - On établit des règles que les agents doivent suivre (rester dans une certaine zone).
- Absence de convention :
  - Les agents peuvent utiliser la communication (directe ou indirecte) pour réaliser un plan en commun.
  - Si un agent commence par exécuter sa partie du plan, les autres doivent respecter cette initiative (reconnaissance du plan initié).
  - Cette "Ethique" fonctionne aussi bien dans le cas d'agents compétitifs que coopératifs.