

---

## Intelligence Artificielle – TD 10

### INTRODUCTION À LA PLANIFICATION

### CORRECTION

---

**Exercice 1** - On se place dans le monde des blocs, décrit dans le cours. On a les actions suivantes :

Action(Deplacer( $b, x, y$ ))

PRECOND :  $\text{Sur}(b, x) \wedge \text{Libre}(b) \wedge \text{Libre}(y) \wedge \text{Bloc}(b) \wedge \text{Bloc}(x) \wedge \text{Bloc}(y) \wedge (b \neq x) \wedge (b \neq y) \wedge (x \neq y)$

EFFET :  $\text{Sur}(b, y) \wedge \text{Libre}(x) \wedge \neg \text{Sur}(b, x) \wedge \neg \text{Libre}(y)$

Action(DeplacerSurTable( $b, x$ ))

PRECOND :  $\text{Sur}(b, x) \wedge \text{Libre}(b) \wedge \text{Bloc}(b) \wedge \text{Bloc}(x) \wedge (b \neq x)$

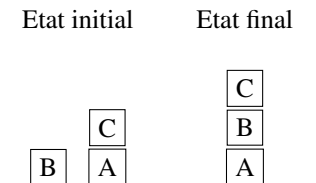
EFFET :  $\text{Sur}(b, \text{Table}) \wedge \text{Libre}(x) \wedge \neg \text{Sur}(b, x)$

Action(DeplacerDeTable( $b, x$ ))

PRECOND :  $\text{Sur}(b, \text{Table}) \wedge \text{Libre}(b) \wedge \text{Libre}(x) \wedge \text{Bloc}(b) \wedge \text{Bloc}(x) \wedge (b \neq x)$

EFFET :  $\neg \text{Sur}(b, \text{Table}) \wedge \neg \text{Libre}(x) \wedge \text{Sur}(b, x)$

Décrivez l'état initial et l'état but du problème suivant, et trouvez une solution en chainage avant permettant de passer de l'un à l'autre.



Indications : N'oubliez pas de vérifier (et de noter) que chaque précondition est bien satisfaite à chaque action appliquée.

**Etat initial :**  $\text{Bloc}(A) \wedge \text{Bloc}(B) \wedge \text{Bloc}(C) \wedge \text{Sur}(B, \text{Table}) \wedge \text{Sur}(A, \text{Table}) \wedge \text{Sur}(C, A) \wedge \text{Libre}(B) \wedge \text{Libre}(C)$

**But :**  $\text{Sur}(A, \text{Table}) \wedge \text{Sur}(B, A) \wedge \text{Sur}(C, B) \wedge \text{Libre}(C)$

*Il n'est pas nécessaire (mais pas interdit non plus) de remettre dans l'état final les prédicats ne pouvant pas changer au cours du temps (comme le fait que A, B et C soient des blocs)*

**Plan totalement ordonné :**

- Etat initial :  $\text{Bloc}(A) \wedge \text{Bloc}(B) \wedge \text{Bloc}(C) \wedge \text{Sur}(B, \text{Table}) \wedge \text{Sur}(A, \text{Table}) \wedge \text{Sur}(C, A) \wedge \text{Libre}(B) \wedge \text{Libre}(C)$
- Action( $\text{DeplacerSurTable}(C, A) - (\{b/C, x/A\})$ )  
 $\text{PRECOND} : \text{Sur}(C, A) \wedge \text{Libre}(C) \wedge \text{Bloc}(C) \wedge \text{Bloc}(A) \wedge (C \neq A)$   
 $\text{EFFETS} : \text{Sur}(C, \text{Table}) \wedge \text{Libre}(A) \wedge \neg \text{Sur}(C, A)$
- Nouvel état:  $\text{Sur}(B, \text{Table}) \wedge \text{Sur}(A, \text{Table}) \wedge \text{Sur}(C, \text{Table}) \wedge \text{Libre}(B) \wedge \text{Libre}(C) \wedge \text{Libre}(A)$
- Action( $\text{DeplacerDeTable}(B, A) - (\{b/B, x/A\})$ )  
 $\text{PRECOND} : \text{Sur}(B, \text{Table}) \wedge \text{Libre}(B) \wedge \text{Libre}(A) \wedge \text{Bloc}(B) \wedge \text{Bloc}(A) \wedge (B \neq A)$   
 $\text{EFFETS} : \neg \text{Sur}(B, \text{Table}) \wedge \neg \text{Libre}(A) \wedge \text{Sur}(B, A)$
- Nouvel état:  $\text{Sur}(B, A) \wedge \text{Sur}(A, \text{Table}) \wedge \text{Sur}(C, \text{Table}) \wedge \text{Libre}(B) \wedge \text{Libre}(C)$
- Action( $\text{DeplacerDeTable}(C, B) - (\{b/C, x/B\})$ )  
 $\text{PRECOND} : \text{Sur}(C, \text{Table}) \wedge \text{Libre}(C) \wedge \text{Libre}(B) \wedge \text{Bloc}(C) \wedge \text{Bloc}(B) \wedge (C \neq B)$   
 $\text{EFFETS} : \neg \text{Sur}(C, \text{Table}) \wedge \neg \text{Libre}(B) \wedge \text{Sur}(C, B)$
- Nouvel état:  $\text{Sur}(B, A) \wedge \text{Sur}(A, \text{Table}) \wedge \text{Sur}(C, B) \wedge \text{Libre}(C)$   
→ tous les prédicats de l'état final sont satisfaits

**Exercice 2** - Un singe mesurant 1 mètre se trouve à l'emplacement A d'une pièce de 2 mètres de hauteur. Une banane est suspendue au plafond, à l'emplacement B de cette pièce, et le singe aimerait bien avoir cette banane. La pièce contient également une caisse, mesurant 1m et placée à l'emplacement C, qu'il peut déplacer et sur laquelle il peut monter.

Nous décrirons l'environnement en STRIPS à l'aide des prédicats suivants :

- $\text{Position}(x, p)$  :  $x$  est à la position  $p$ , où  $p$  peut être égal à  $A, B$  ou  $C$
- $\text{Hauteur}(x, h)$  :  $x$  est à la hauteur  $h$ , où  $h$  peut-être *bas* ou *haut*
- $\text{PeutPousser}(c)$  : le singe est capable de pousser  $c$
- $\text{PeutGrimper}(c)$  : le singe est capable de grimper sur  $c$
- $\text{PeutAttraper}(b)$  : le singe est capable d'attraper  $b$
- $\text{Sur}(c)$  : le singe est sur  $c$
- $\text{Tient}(b)$  : le singe tient  $b$

Nous utiliserons en outre trois constantes : *Singe*, *Caisse* et *Banane*.

Le singe est capable d'effectuer 6 actions :

- $\text{Aller}(x, y)$  : le singe se déplace de l'emplacement  $x$  à l'emplacement  $y$ . Pour pouvoir marcher, il doit être sur le sol (et donc en bas)
- $\text{Pousse}(c, x, y)$  : le singe pousse  $c$  de l'emplacement  $x$  à l'emplacement  $y$ . Dans ce cas, il doit également être sur le sol pour pouvoir se déplacer en poussant  $c$ .
- $\text{Attrape}(b, x, h)$  : le singe attrape  $b$ , qui est en position  $x$  et à la hauteur  $h$ . A l'issue de cette action, le singe tient  $b$
- $\text{Grimpe}(c, x)$  : le singe grimpe sur  $c$  qui est en position  $x$ .
- $\text{Descend}(c)$  : le singe, qui était sur  $c$ , en descend
- $\text{Lache}(b, h)$  : le singe lâche  $b$  qu'il tenait. Quelle que soit la hauteur  $h$  initiale de  $b$ ,  $b$  se retrouve en bas.

1. Donner en STRIPS l'état initial et l'état but de ce problème

**Etat initial :**

$\text{Position}(\text{Singe}, A) \wedge \text{Position}(\text{Banane}, B) \wedge \text{Position}(\text{Caisse}, C) \wedge \text{Hauteur}(\text{Singe}, \text{Bas}) \wedge \text{Hauteur}(\text{Banane}, \text{Haut}) \wedge \text{Hauteur}(\text{Caisse}, \text{Bas}) \wedge \text{PeutPousser}(\text{Caisse}) \wedge \text{PeutGrimper}(\text{Caisse}) \wedge \text{PeutAttraper}(\text{Banane})$

**But :**  $\text{Tient}(\text{Banane})$

## 2. Décrivez les actions nécessaires au singe pour atteindre son objectif

```
Action(Aller(p1, p2)
  PRECOND : Position(Singe, p1) ∧ Hauteur(Singe, Bas)
  EFFET : Position(Singe, p2) ∧ ¬Position(Singe, p1))

Action(Pousse(c, p1, p2)
  PRECOND : PeutPousser(c) ∧ Position(c, p1) ∧ Position(Singe, p1) ∧ Hauteur(Singe, Bas)
  EFFET : Position(c, p2) ∧ Position(Singe, p2) ∧ ¬Position(c, p1) ∧ ¬Position(Singe, p1))

Action(Grimpe(c, p)
  PRECOND : PeutGrimper(c) ∧ Position(c, p) ∧ Position(Singe, p) ∧ Hauteur(c, Bas) ∧
    Hauteur(Singe, Bas)
  EFFET : Sur(c) ∧ ¬Hauteur(Singe, Bas) ∧ Hauteur(Singe, Haut))

Action(Attrape(b, p, h)
  PRECOND : Position(b, p) ∧ Position(Singe, p) ∧ Hauteur(b, h) ∧ Hauteur(Singe, h) ∧
    PeutAttraper(b)
  EFFET : Tient(b))

Action(Descend(c)
  PRECOND : Sur(c) ∧ Hauteur(Singe, Haut)
  EFFET : ¬ Sur(c) ∧ ¬Hauteur(Singe, Haut) ∧ Hauteur(Singe, Bas))

Action(Lache(b, h)
  PRECOND : Tient(b)
  EFFET : ¬Tient(b) ∧ ¬Hauteur(b, Haut) ∧ Hauteur(b, Bas))
```

3. Trouvez un plan totalement ordonné en chaînage avant permettant de passer de l'état initial à l'état qui satisfait le but

- Etat initial :

Position(Singe, A)  $\wedge$  Position(Banane, B)  $\wedge$  Position(Caisse, C)  $\wedge$  Hauteur(Singe, Bas)  $\wedge$  Hauteur(Banane, Haut)  $\wedge$  Hauteur(Caisse, Bas)  $\wedge$  PeutPousser(Caisse)  $\wedge$  PeutGrimper(Caisse)  $\wedge$  PeutAttraper(Banane)

- Action(Aller(A, B) – ({p1/A, p2/C}))

PRECOND : Position(Singe, A)  $\wedge$  Hauteur(Singe, Bas)

EFFETS : Position(Singe, C)  $\wedge$   $\neg$ Position(Singe, A)

- Nouvel état :

Position(Singe, C)  $\wedge$  Position(Banane, B)  $\wedge$  Position(Caisse, C)  $\wedge$  Hauteur(Singe, Bas)  $\wedge$  Hauteur(Banane, Haut)  $\wedge$  Hauteur(Caisse, Bas)  $\wedge$  PeutPousser(Caisse)  $\wedge$  PeutGrimper(Caisse)  $\wedge$  PeutAttraper(Banane)

- Action(Pousse(Caisse, C, B) – ({c/Caisse, p1/C, p2/B}))

PRECOND : PeutPousser(Caisse)  $\wedge$  Position(Caisse, C)  $\wedge$  Position(Singe, C)  $\wedge$  Hauteur(Singe, Bas)

EFFETS : Position(Caisse, B)  $\wedge$  Position(Singe, B)  $\wedge$   $\neg$ Position(Caisse, C)  $\wedge$   $\neg$ Position(Singe, C)

- Nouvel état :

Position(Singe, B)  $\wedge$  Position(Banane, B)  $\wedge$  Position(Caisse, B)  $\wedge$  Hauteur(Singe, Bas)  $\wedge$  Hauteur(Banane, Haut)  $\wedge$  Hauteur(Caisse, Bas)  $\wedge$  PeutPousser(Caisse)  $\wedge$  PeutGrimper(Caisse)  $\wedge$  PeutAttraper(Banane)

- Action(Grimpe(Caisse, B) – ({c/Caisse, p/B}))

PRECOND : PeutGrimper(Caisse)  $\wedge$  Position(Caisse, B)  $\wedge$  Position(Singe, B)  $\wedge$  Hauteur(Caisse, Bas)  $\wedge$  Hauteur(Singe, Bas)

EFFETS : Sur(Caisse)  $\wedge$   $\neg$ Hauteur(Singe, Bas)  $\wedge$  Hauteur(Singe, Haut)

- Nouvel état :

Position(Singe, B)  $\wedge$  Position(Banane, B)  $\wedge$  Position(Caisse, B)  $\wedge$  Hauteur(Singe, Haut)  $\wedge$  Hauteur(Banane, Haut)  $\wedge$  Hauteur(Caisse, Bas)  $\wedge$  PeutPousser(Caisse)  $\wedge$  PeutGrimper(Caisse)  $\wedge$  PeutAttraper(Banane)  $\wedge$  Sur(Caisse)

- Action(Attrape(Banane, B, Haut) – ({b/Banane, p/B, h/Haut}))

PRECOND : Position(Banane, B)  $\wedge$  Position(Singe, B)  $\wedge$  Hauteur(Banane, Haut)  $\wedge$  Hauteur(Singe, Haut)  $\wedge$  PeutAttraper(Banane)

EFFETS : Tient(Banane)

- Nouvel état :

Position(Singe, B)  $\wedge$  Position(Banane, B)  $\wedge$  Position(Caisse, B)  $\wedge$  Hauteur(Singe, Haut)  $\wedge$  Hauteur(Banane, Haut)  $\wedge$  Hauteur(Caisse, Bas)  $\wedge$  PeutPousser(Caisse)  $\wedge$  PeutGrimper(Caisse)  $\wedge$  PeutAttraper(Banane)  $\wedge$  Sur(Caisse)  $\wedge$  Tient(Banane)

→ tous les prédicats de l'état final sont satisfaits

**Exercice 3** - On se place dans une pièce que l'on aimerait repeindre. A l'état initial, on a une échelle et des pinceaux, mais il nous manque de la peinture. D'autre part, la lampe (qui est au sol) ne fonctionne pas, on ne peut pas peindre sans lumière, et on a pas d'ampoule de rechange. A l'état final, on voudrait avoir peint et avoir de la lumière dans cette pièce.

Nous décrivons l'environnement en STRIPS à l'aide des prédicats suivants :

- Objet ( $o$ ) :  $o$  est un objet
- Avoir ( $o$ ) : être en possession d'un objet  $o$
- Lampe ( $l$ ) :  $l$  est une lampe
- Etat ( $l, e$ ) : décrit l'état de  $l$ . Deux états sont possibles :  $e = \text{Cassée}$  ou  $e = \text{Fonctionne}$
- Pièce ( $p$ ) :  $p$  est une pièce
- Eclairée ( $p$ ) :  $p$  est éclairée
- Peinte ( $P$ ) :  $p$  est peinte

Nous avons 4 actions possible :

- Peindre ( $p$ ), où  $p$  est la pièce à peindre. Pour pouvoir peindre cette pièce, il faut avoir de la peinture, des pinceaux, et une échelle pour atteindre le plafond. La pièce doit en outre être éclairée. On suppose qu'une fois la pièce peinte, il n'y a plus de peinture.
- Changer-Ampoule ( $l$ ), où  $l$  est une lampe qui ne fonctionne pas. Il faut avoir une ampoule en stock, et la lampe fonctionne une fois cette action effectuée.
- Allumer ( $p, l$ ), où  $p$  est une pièce et  $l$  une lampe qui fonctionne. Cette action permet d'éclairer la pièce  $p$
- Obtenir ( $o$ ), où  $o$  est un objet. Cette action, qui abstrait une suite d'actions élémentaires dont on ne s'occupe pas ici, permet d'obtenir l'objet  $o$ .

#### 1. Donner en STRIPS l'état initial et le but de ce problème

**Etat initial :** Avoir(Echelle)  $\wedge$  Avoir(Pinceaux)  $\wedge$  Lampe(L)  $\wedge$  Etat(L, Cassée)  $\wedge$  Pièce(P)  
 $\wedge$  Objet(Echelle)  $\wedge$  Objet(Pinceaux)  $\wedge$  Objet(Amp)  $\wedge$  Objet(Pture)  
**But :** Peinte(P)  $\wedge$  Eclairée(P)

#### 2. Décrire les actions Peindre( $x$ ), Changer-Ampoule( $x$ ), Allumer( $x, y$ ), Obtenir( $x$ )

```

Action(Peindre(x)
  PRECOND : Pièce(x)  $\wedge$  Avoir(Echelle)  $\wedge$  Avoir(Pinceaux)  $\wedge$  Avoir(Pture)  $\wedge$ 
            Eclairée(x)
  EFFET : Peinte(x)  $\wedge$   $\neg$ Avoir(Pture))

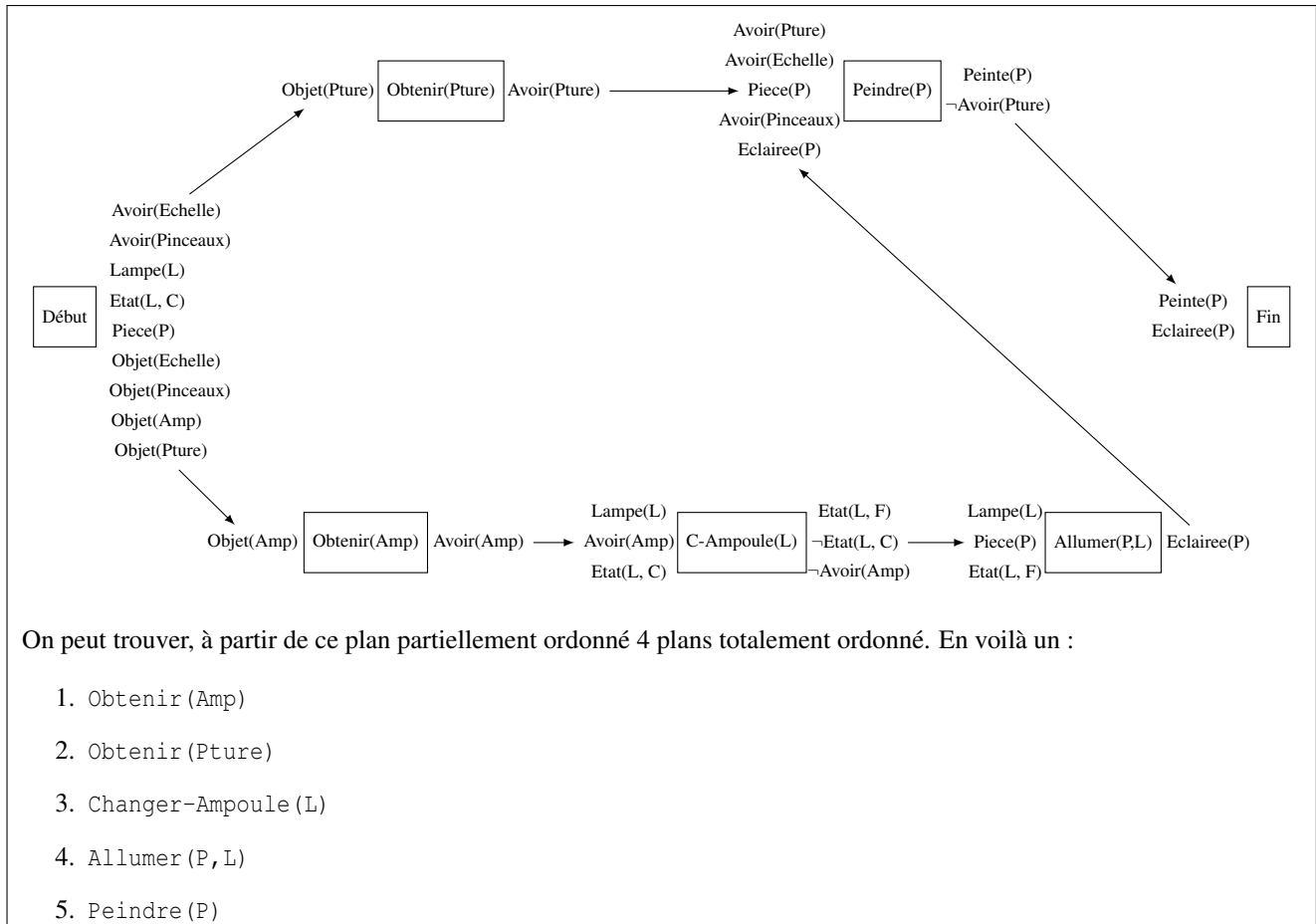
Action(Changer-ampoule(x)
  PRECOND : Lampe(x)  $\wedge$  Avoir(Ampoule)  $\wedge$  Etat(x, Cassée)
  EFFET : Etat(x, Fonctionne)  $\wedge$   $\neg$ Avoir(Ampoule)  $\wedge$   $\neg$ Etat(x, Cassée))

Action(Allumer(x, y)
  PRECOND : Pièce(x)  $\wedge$  Lampe(y)  $\wedge$  Etat(y, Fonctionne)
  EFFET : Eclairée(x))

Action(Obtenir(x)
  PRECOND : Objet(x)
  EFFET : Avoir(x))

```

3. Trouver un plan partiellement ordonné permettant de résoudre ce problème. Donner ensuite une solution permettant de passer de l'état initial à l'état final



On peut trouver, à partir de ce plan partiellement ordonné 4 plans totalement ordonné. En voilà un :

1. Obtenir(Amp)
2. Obtenir(Pture)
3. Changer-Ampoule(L)
4. Allumer(P, L)
5. Peindre(P)