

# IFT-17586

## Intelligence artificielle I



### Planification

Benoît Potvin

Département d'informatique et de génie logiciel  
Faculté des sciences et de génie, Université Laval  
Été 2003

## La planification



### ◆ Référence

- G.F. Luger, Artificial Intelligence. Structures and Strategies for Complex Problem Solving, Addison Wesley, (Mass.), 4e éd., 2002, ch. 7, pp. 284-293.

# La planification

## STRIPS

© Tourigny, Potvin et Capus (2003)

3

## STRIPS

- ◆ STRIPS : STandford Research Institute Planning System (1971)
  - Représenter et implémenter efficacement les opérations d'un planificateur
  - Une solution au problème des schémas (**frame problem**)
  - Apprentissage

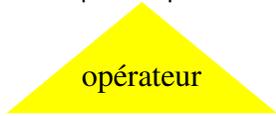
© Tourigny, Potvin et Capus (2003)

4

# STRIPS

## ◆ Opérateurs

Liste **P** :  
Éléments qui **doivent être vrais** pour  
que l'opérateur puisse s'appliquer



Liste **A** :  
éléments à **Ajouter** à  
l'état suite à l'application  
de l'opérateur

Liste **S** :  
éléments à **Supprimer** de  
l'état suite à l'application  
de l'opérateur

La liste **S** spécifie tout ce qui est nécessaire  
pour satisfaire les axiomes de schémas

# STRIPS

## ◆ Exemple : Le monde des blocs

Logique des prédicats

7.  $(\forall X) (\forall Y) (\text{depiler}(X,Y) \rightarrow$   
 $((\text{libre}(Y) \wedge \text{tenir}(X)) \leftarrow$   
 $(\text{sur}(X,Y) \wedge \text{tenir}() \wedge \text{libre}(X))))$

8.  $(\forall X) (\forall Y) (\forall Z) (\text{depiler}(Y,Z) \rightarrow$   
 $(\text{surtable}(X) \leftarrow \text{surtable}(X)))$

11.  $(\forall X) (\forall Y) (\forall Z) (\forall W) (\text{depiler}(Z,W) \rightarrow$   
 $(\text{sur}(X,Y) \leftarrow (\text{sur}(X,Y) \wedge X \neq Z)))$

12.  $(\forall X) (\forall Y) ((\text{tenir}(Y) \vee \text{tenir}()) \rightarrow$   
 $(\text{libre}(X) \leftarrow \text{libre}(X)))$

STRIPS

Préconditions :  
 $\text{sur}(X,Y) \wedge \text{tenir}() \wedge \text{libre}(X)$



Ajouter :  $\text{libre}(Y) \wedge \text{tenir}(X)$       Supprimer :  $\text{tenir}() \wedge \text{sur}(X,Y)$

# STRIPS

## ◆ Exemple : le monde des blocs

|              |  |
|--------------|--|
| depiler(X,Y) | P : [ tenir(), libre(X), sur(X,Y) ]<br>A : [ tenir(X), libre(Y) ]<br>S : [ tenir(), sur(X,Y) ] |
|--------------|--|

|            |  |
|------------|--|
| deposer(X) | P : [ tenir(X) ]<br>A : [ tenir(), surtable(X), libre(X) ]<br>S : [ tenir(X) ] |
|------------|--|

...

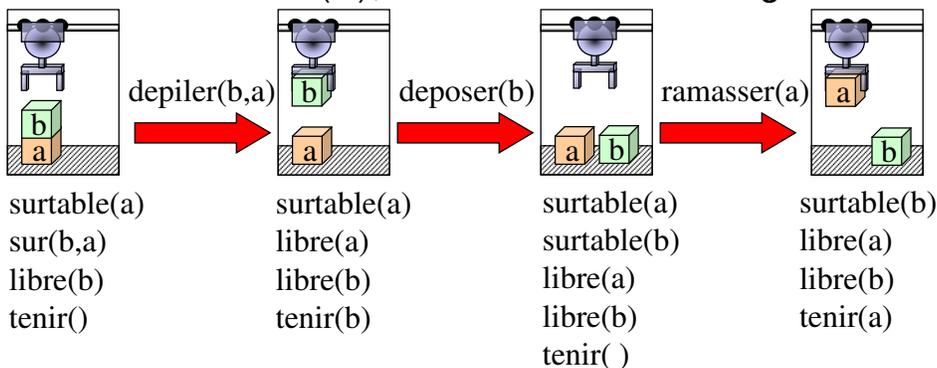
© Tourigny, Potvin et Capus (2003)

7

# STRIPS

## ◆ Exemple : le monde des blocs

• but : tenir(a), état initial : 1er à gauche



© Tourigny, Potvin et Capus (2003)

8

# STRIPS

## ◆ Construire un plan

- Spécifier les opérations nécessaires pour atteindre un but
- Diviser le but en sous-buts et construire un plan pour les atteindre
  - Résoudre les sous-buts indépendamment peut causer des problèmes si un sous-but contient une opération qui annule l'opération d'un autre sous-but

© Tourigny, Potvin et Capus (2003)

9

# STRIPS

## ◆ Macro-opérateurs

- Suite d'opérateurs permettant d'atteindre un but / un sous-but
- Sorte de plan / sous-plan
- Listes P, A, S
- Comment représenter efficacement les macro-opérateurs?
  - Tables triangle (***triangle tables***)

© Tourigny, Potvin et Capus (2003)

10

# STRIPS

- ◆ Table triangulaire (***Triangle table***)
  - Structure de données pour organiser une séquence d'actions
  - Principe : relier les préconditions d'une action aux postconditions des actions qui la précèdent
  - Si un plan fonctionne, le généraliser et le sauvegarder sous la forme d'un macro-opérateur

© Tourigny, Potvin et Capus (2003)

11

# STRIPS

- ◆ Table triangulaire (***Triangle table***)

**Opération # n**

**Rangée #n :**

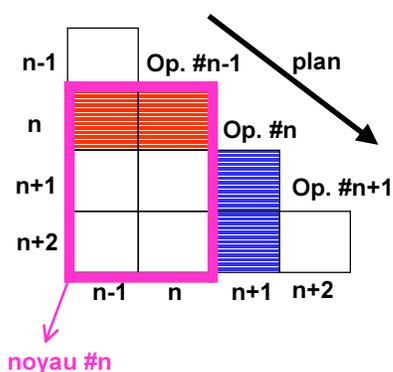
Préconditions de l'opération #n

**Colonne #n+1:**

Postconditions de l'opération #n

**Noyau #n :**

Si un problème survient, l'opération #n ne peut être effectuée que si toutes les préconditions dans le noyau #n sont vérifiées



© Tourigny, Potvin et Capus (2003)

12

# STRIPS

## ◆ Exemple : le monde des blocs

But: tenir(a)

Plan:

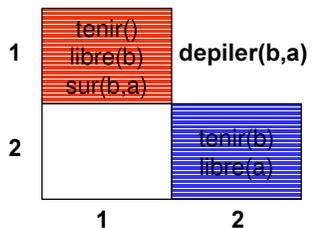
1. **depiler(b,a)**
2. déposer(b)
3. ramasser(a)

Préconditions :

tenir(), libre(b), sur(b,a)

Postconditions :

tenir(b), libre(a)



© Tourigny, Potvin et Capus (2003)

13

# STRIPS

## ◆ Exemple : le monde des blocs

But: tenir(a)

Plan:

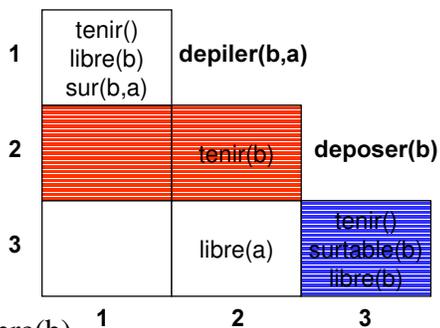
1. depiler(b,a)
2. **deposer(b)**
3. ramasser(a)

Préconditions :

tenir(b)

Postconditions :

tenir(), surtable(b), libre(b)



© Tourigny, Potvin et Capus (2003)

14

# STRIPS

## ◆ Exemple : le monde des blocs

But: tenir(a)

Plan:

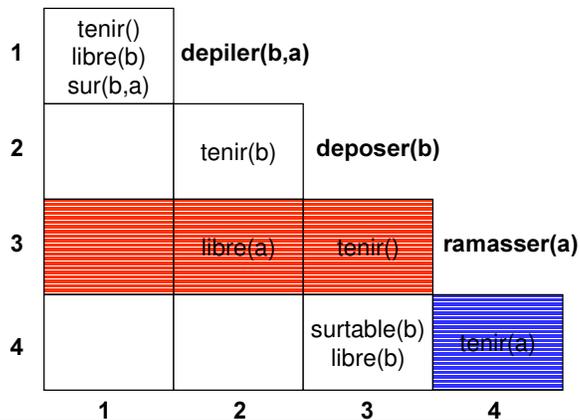
1. depiler(b,a)
2. déposer(b)
3. ramasser(a)

Préconditions :

libre(a), tenir()

Postconditions :

tenir(a)



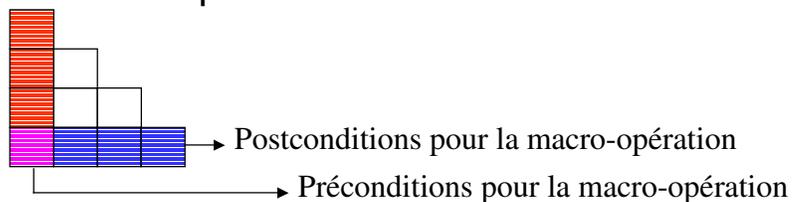
© Tourigny, Potvin et Capus (2003)

15

# STRIPS

## ◆ Macro-opération

- Généraliser : remplacer les constantes par des variables
- Préconditions et postconditions de la macro-opération



© Tourigny, Potvin et Capus (2003)

16

# STRIPS

## ◆ Exemple : le monde des blocs

### – Macro

But: tenir(a)

Préconditions :

tenir()

sur(b,a)

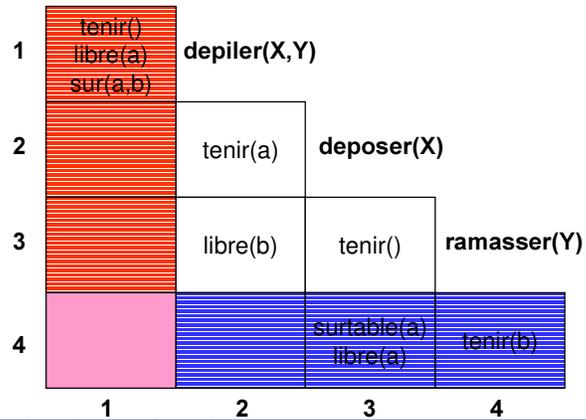
libre(b)

Postconditions:

surtable(b)

libre(b)

tenir(a)



© Tourigny, Potvin et Capus (2003)

17

# STRIPS

## ◆ Apprentissage

– Généraliser et sauvegarder les plans ayant réussi

– Réutiliser ces plans dans des situations similaires

© Tourigny, Potvin et Capus (2003)

18

# STRIPS

## ◆ Exemple : le monde des blocs

### – Généraliser

But: tenir(Y)

Préconditions :

tenir()

sur(X,Y)

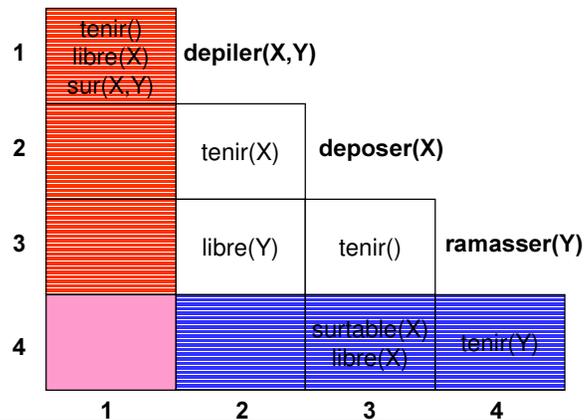
libre(X)

Postconditions:

surtable(X)

libre(X)

tenir(Y)



© Tourigny, Potvin et Capus (2003)

19

# STRIPS

## ◆ Avantages des Macros

- Résoudre efficacement des sous-buts possiblement conflictuels
- Apprentissage

## ◆ Limites des Macros

- Perte d'une partie de la sémantique
  - Les seules postconditions d'une opérations sont celles qui peuvent servir de préconditions à une autre
- Croissance du nombre d'opérateurs

© Tourigny, Potvin et Capus (2003)

20

## Autres techniques de planification

### ◆ Planification téléo-réactive

- ref : Luger, 2002, pp. 293-296
- Indépendant du domaine
- Mise à profit d'agents

### ◆ Planification à base de modèle

- ref : Luger, 2002, pp. 296-299
- Utilisation d'un modèle du comportement du système
- facteurs : coût d'une opération normale, probabilité qu'un état soit anormal