

**4º Trabalho de Inteligência Artificial**  
**2018/2019**



**UNIVERSIDADE  
DE ÉVORA**

**Planeamento**

Docente:

Irene Pimenta Rodrigues

Realizado por:

Daniel Serrano – 35087

Miguel Serrano – 34149

## 1. Trabalho

Considere que tem um robot com uma mão que pode abrir e fechar as gavetas de um modulo (uma de cada vez porque só tem uma mão), pegar nos objetos que estão nas gavetas abertas (um de cada vez) e colocar o objeto que tem na mão numa gaveta aberta. Note que quando o robot tem um objeto na mão não é capaz de abrir uma gaveta mas consegue fecha-la; e que só consegue colocar um objeto numa gaveta aberta se a gaveta de cima estiver fechada. Suponha que o modulo tem 3 gavetas e há 3 objetos: 'a' na gaveta 1, 'b' na gaveta 2 e 'c' na gaveta 3.

## 2. Exercício 1 - Notação STRIPS (Condições e ações)

### 2.1. Abrir gaveta

```
%abrir gaveta  
accao(abreGav(G), [mao([]), fechada(G)], [aberta(G)], [fechada(G)]) :- member(G, [1,2,3]).
```

Abre a gaveta “G”. A mão do robot tem de estar vazia para utilizar esta ação. Após a utilização desta ação o estado da gaveta passa de “fechada” para “aberta”.

### 2.2. Fecha gaveta

```
%fechar gaveta  
accao(fechaGav(G), [aberta(G)], [fechada(G)], [aberta(G)]) :- member(G, [1,2,3]).
```

Fecha a gaveta “G”. Após a utilização desta ação o estado da gaveta passa de “aberta” para “fechada”.

### 2.3. Coloca objeto

```
%coloca objeto  
accao(colocaObj(G,O), [mao(O), aberta(G), fechada(G1)], [conteudo(G,O), mao([]), [mao(O)]] :- seguinte(G1,G), member(O, [a,b,c]).
```

Coloca o objeto “O” que está em posse da mão do robot, na gaveta “G”. Para que esta ação seja feita, a gaveta “G” tem de estar “aberta”, a gaveta seguinte da gaveta “G” tem de estar “fechada” e o robot estar na posse do objeto “O”.

## 2.4. Retira objeto

```
%retira objeto  
acciao(tiraObj(G,O), [mao([]), aberta(G), conteudo(G,O), fechada(G1)], [mao(O)], [conteudo(G,O), mao([])]):- seguinte(G1,G), member(O, [a,b,c]).
```

Retira o objeto “O” da gaveta “G”. Para que seja feita esta ação, a gaveta “G” tem de estar “aberta”, a gaveta seguinte da gaveta “G” estar “fechada” e a gaveta “G” conter o objeto “O”.

## 2.5. Condições

- > mao(O);
- > fechada(G);
- > aberta(G);
- > conteudo(G,O).

## 2.6. Ações

- > abreGav(G);
- > fechaGav(G).
- > tiraObj(G,O);
- > colocaObj(G,O);

### 3. Estados

#### Estado 0 (estado inicial):

```
%estado inicial - 0
estado_inicial([mao([]), conteudo(1,a), conteudo(2,b), conteudo(3,c), fechada(1), fechada(2), fechada(3), fechada(0)] ).
```

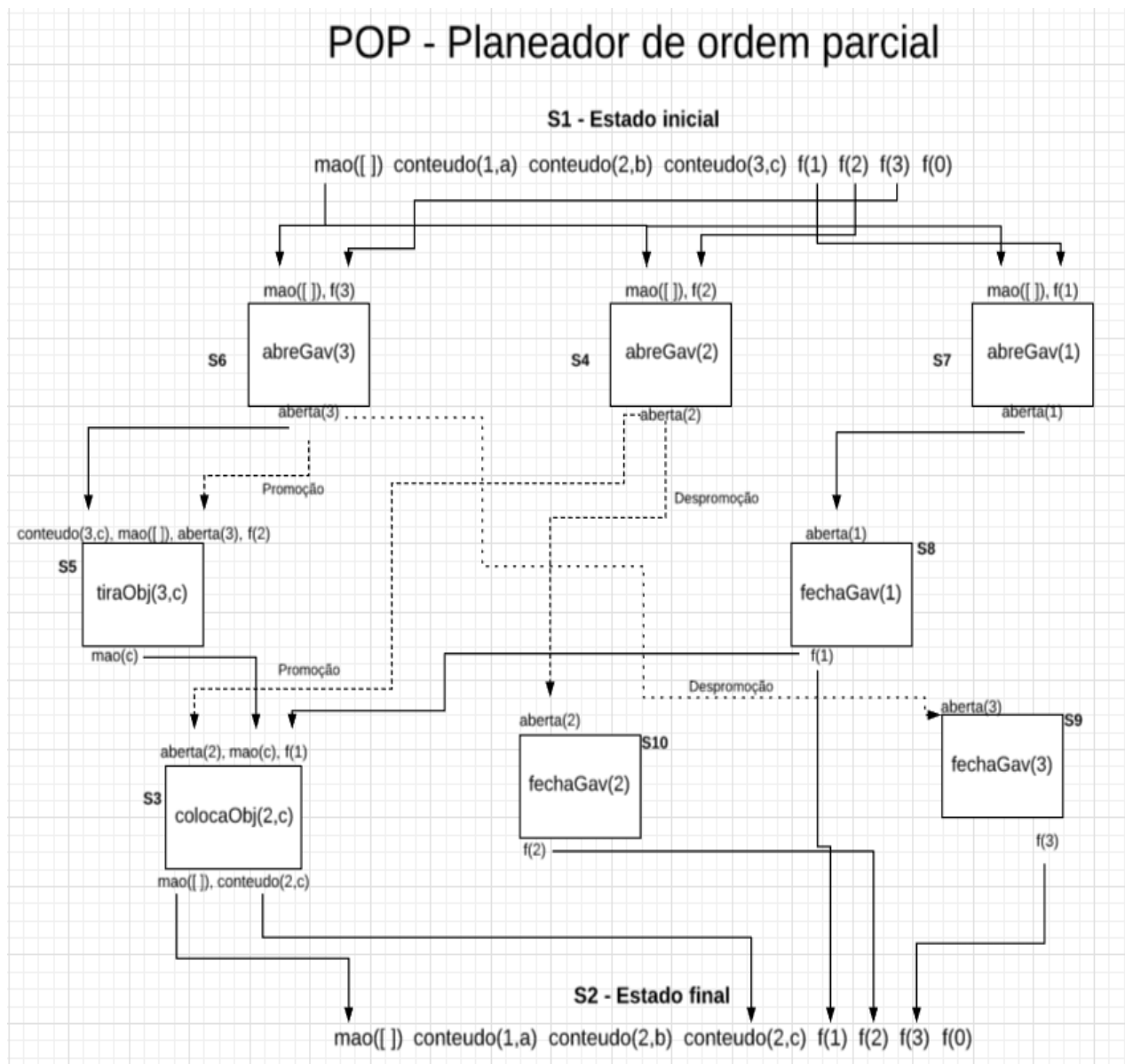
#### Estado 1 (estado final):

```
%estado final - 1
estado_1([mao([]), conteudo(1,a), conteudo(2,b), conteudo(2,c), fechada(1), fechada(2), fechada(3), fechada(0)] ).
```

#### Estado 2(estado final):

```
%estado final - 2
estado_2([mao([]), conteudo(2,a), conteudo(2,c), conteudo(3,b), fechada(1), fechada(2), fechada(3), fechada(0)] ).
```

## 4. POP – Planeador de ordem parcial



## 5. Links e ataques

-> S4 para S3(aberta(2)). S10 ataca o link de S4 para S3, S10 terá de se realizar após S3, logo acontece a despromoção.

-> S6 para S5(aberta(3)). S9 ataca o link de S6 para S5, S9 terá de se realizar após S5, logo acontece a despromoção.

## 6. Sequência de ações

De forma a conseguir-se ver resultados com uma resolução menos demorada, criei o seguinte estado final alternativo:

```
%estado final alternativo  
estado_final([conteudo(3,a), conteudo(2,b), conteudo(3,c), fechada(1), fechada(2), fechada(3)]).
```

Com a seguinte sequência de ações:

```
% Plano = [s1-inicial,s243-abreGav(1),s244-abreGav(3),s242-tiraObj(1,a),s241-colocaObj(3,a),s245-fechaGav(1),s246-fechaGav(3),s2-final]
```