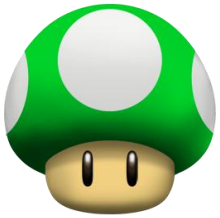
INF 1771 – Inteligência Artificial

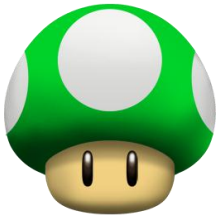
Aula 11 – Planejamento

Edirlei Soares de Lima
<elima@inf.puc-rio.br>



Introdução

- Agentes vistos anteriormente:
  - **Agentes baseados em busca.**
    - Busca cega;
    - Busca heurística;
    - Busca local;
  - **Agentes baseados em lógica.**
    - Lógica proposicional;
    - Lógica de primeira ordem;
    - Prolog;



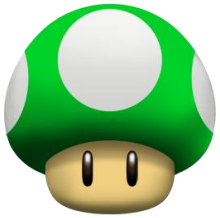
Introdução

- **Planejamento** consiste na tarefa de apresentar uma sequência de ações para alcançar um determinado objetivo.

  Ir(Mercado), Comprar(Biscoito), Ir(Farmácia), Comprar(Remédio), Ir(Casa)

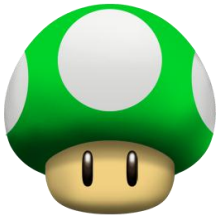
- Dado um objetivo, um **agente planejador** deve ser capaz de construir um plano de ação para chegar ao seu objetivo.

- Após planejar, o agente deve **executar as ações do plano** uma a uma.



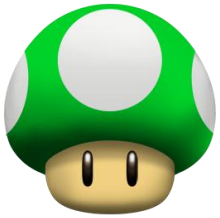
Funcionamento de um Agente Planejador

- Inicialmente um agente planejador **gera um objetivo** a alcançar.
- **Constrói um plano** para atingir o objetivo a partir do estado atual do ambiente.
- **Executa o plano** do começo ao fim.
- **Gera um novo objetivo** com base no novo estado do ambiente.



Planejamento

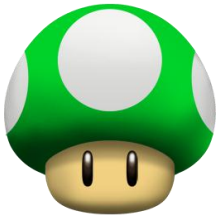
- Em **planejamento clássico** o ambiente do problema possui as seguintes características:
  - Observável
  - Determinístico
  - Finito
  - Estático



Resolução de Problemas X Planejamento

- **Algoritmos de busca** tendem a tomar ações irrelevantes.
  - Grande fator de ramificação.
  - Pouco conhecimento para guiar a busca.

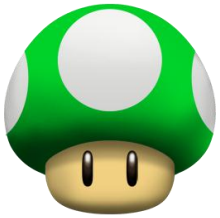
- **Planejador** não considera ações irrelevantes.
  - Faz conexões diretas entre estados (sentenças) e ações (pré-condições + efeitos)
  - Objetivo: Ter(Leite).
    - Ação: Comprar(Leite) => Ter(Leite)



Resolução de Problemas X Planejamento

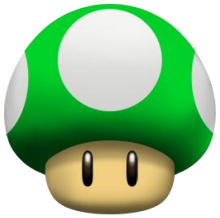
- Em problemas do mundo real é difícil definir uma boa heurística para **algoritmos de busca heurística**.

- Um **planejador** tem acesso a representação explícita do objetivo.
  - Objetivo: conjunção de sub-objetivos que levam ao objetivo final.
  - Heurística **única**: número de elementos da conjunção não-satisfeitos.



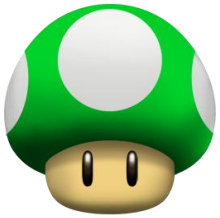
Resolução de Problemas X Planejamento

- **Algoritmos de busca** não tiram proveito da decomposição do problema.
- **Planejadores** aproveitam a estrutura do problema. É possível decompor com facilidade sub-objetivos.
  - Exemplo: Ter(A) Λ Ter(B) Λ Ter(C) Λ Ter(D)



Linguagem STRIPS

- **Linguagem formal** para a especificação de problemas de planejamento.
- **Representação de estados**: conjunção de literais positivos sem variáveis.
  - **Inicial:** Em(Casa)
  - **Final:** Em(Casa) ^ Ter(Leite) ^ Ter(Bananas) ^ Ter(Furadeira)
  - **Hipótese do mundo fechado:** qualquer condição não mencionada em um estado é considerada negativa.
    - Exemplo: ¬Ter(Leite) ^ ¬Ter(Bananas) ^ ¬Ter(Furadeira)

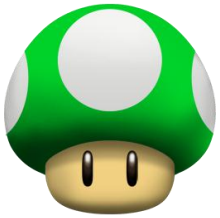


Linguagem STRIPS

- **Objetivos:** conjunção de literais e possivelmente variáveis:
  - Em(Casa) ^ Ter(Leite) ^ Ter(Bananas) ^ Ter(Furadeira)
  - Em(x) ^ Vende(x, Leite)

- **Ações** são especificadas em termos de pré-condições e efeitos:

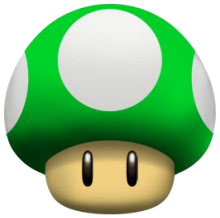
  - **Descritor da ação:** predicado lógico
  - **Pré-condição:** conjunção de literais positivos
  - **Efeito:** conjunção de literais (positivos ou negativos)



Linguagem STRIPS

- Operador para ir de um lugar para outro:

Ação(Ir(Destino),
Pré-condição Em(Partida) ^ Caminho(Partida, Destino),
Efeito Em(Destino) ^ ¬ Em(Partida))



Exemplo – Transporte Aéreo de Carga

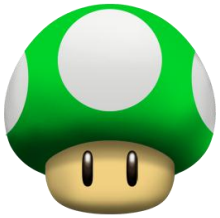
Início(Em(C1, SFO) ^ Em(C2,JFK) ^ Em(A1,SFO) ^ Em(A2,JFK) ^ Carga(C1) ^ Carga(C2) ^ Avião(A1) ^ Avião(A2) ^ Aeroporto(JFK) ^ Aeroporto(SFO))

Objetivo(Em(C1,JFK) ^ Em(C2,SFO))

Ação(**Carregar**(c,a,l)
PRÉ-CONDIÇÃO: Em(c,l) ^ Em(a,l) ^ Carga(c) ^ Avião(a) ^ Aeroporto(l)
EFEITO: ¬Em(c,l) ^ Dentro(c,a))

Ação(**Descarregar**(c,a,l)
PRÉ-CONDIÇÃO: Dentro(c,a) ^ Em(a,l) ^ Carga(c) ^ Avião(a) ^ Aeroporto(l)
EFEITO: Em(c,l) ^ ¬Dentro(c,a))

Ação(**Voar**(a,de,para)
PRÉ-CONDIÇÃO: Em(a,de) ^ Avião(a) ^ Aeroporto(de) ^ Aeroporto(para)
EFEITO: ¬ Em(a,de) ^ Em(a,para))

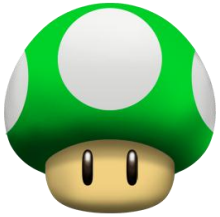


Tipos de Planejadores

- Formas de Buscas de Planos:
  - **Progressivo:** estado inicial -> objetivo.
  - **Regressivo:** objetivo -> estado inicial.
    - mais eficiente (há menos caminhos partindo do objetivo do que do estado inicial)

- Espaços de busca:
  - **Espaço de situações:** Funciona da mesma forma que na resolução de problemas por meio de busca.

  - **Espaço de planos:** planos parciais.
    - mais flexível.

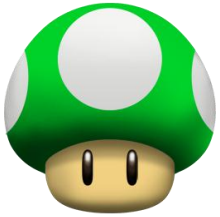


Busca em Espaço de situações

- A **busca em espaço de situações** é **ineficiente** devido a ela não considerar o problema das ações irrelevantes. Todas as opções de ações são testadas em cada estado.

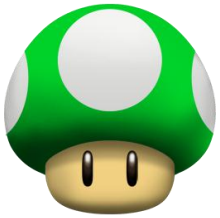
- Isso faz com que a complexidade do problema cresça muito rapidamente.

- **Solução?** Busca no espaço de planos parciais **(planejamento de ordem parcial)**.



Planejamento de Ordem Parcial

- **Subdivisão do problema**.
- **Ordem de elaboração do plano flexível**.
- **Compromisso mínimo.**
  - Adiar decisões durante a procura.
- O planejador de ordem parcial pode inserir duas ações em um plano sem especificar qual delas deve ser executada primeiro.



Exemplo dos Sapatos

Inicio()

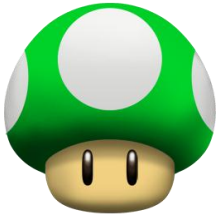
Objetivo(SapatoDireitoCalçado^SapatoEsquerdoCalçado)

Ação(SapatoDireito,
PRECOND: MeiaDireitaCalçada,
EFFECT: SapatoDireitoCalçado)

Ação(MeiaDireita,
EFFECT: MeiaDireitaCalçada)

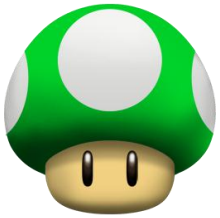
Ação(SapatoEsquerdo,
PRECOND: MeiaEsquerdaCalçada,
EFFECT: SapatoDireitoCalçado)

Ação(MeiaEsquerda,
EFFECT: MeiaEsquerdaCalçada)



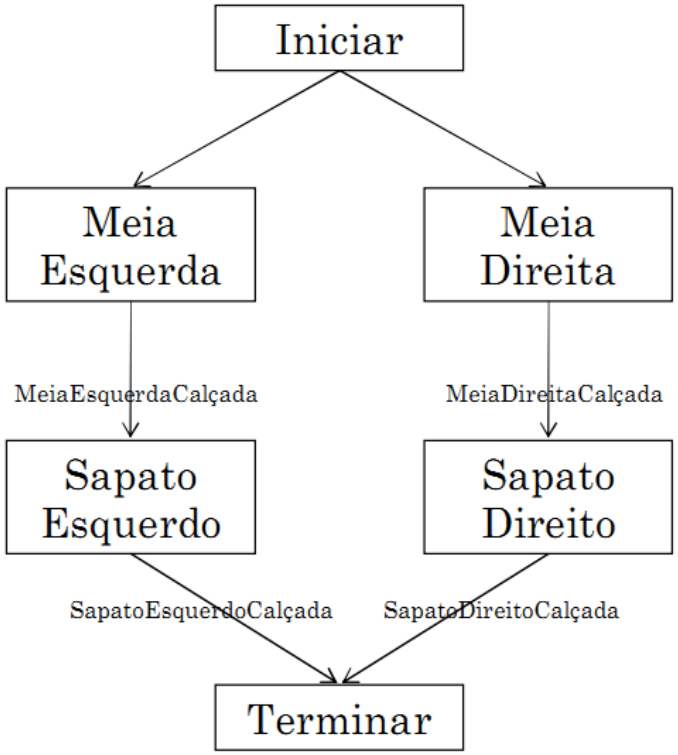
Exemplo dos Sapatos

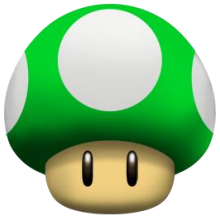
- Um planejador de ordem parcial deve ser capaz de chegar a **duas sequências de ações**:
  - MeiaDireita seguido por SapatoDireito;
  - MeiaEsqueda seguido por SapatoEsquerdo.
- As duas sequências podem ser **combinadas** para produzir o plano final.



Exemplo dos Sapatos

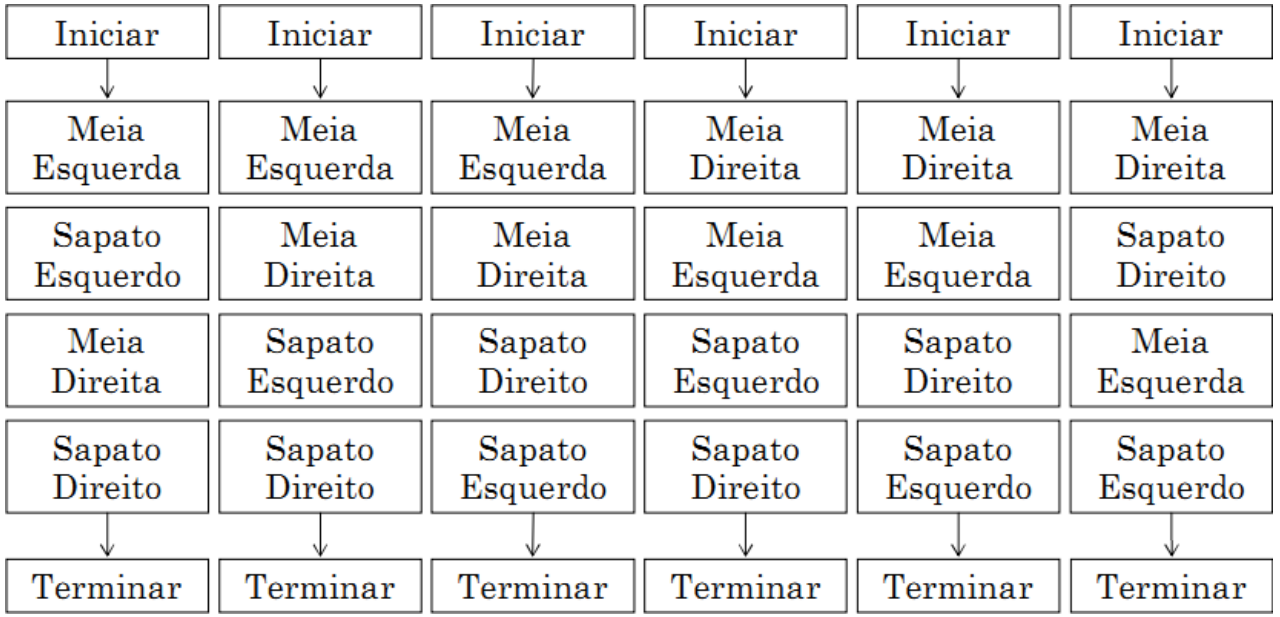
- Plano de Ordem Parcial

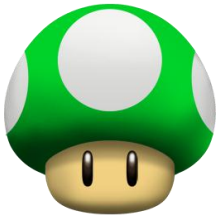




Exemplo dos Sapatos

- Plano de Ordem Total



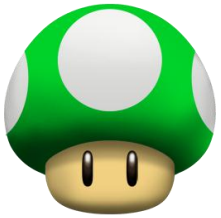


Planejamento de Ordem Parcial

- O planejamento de ordem parcial pode ser implementado como uma **busca no espaço de ordem parcial de planos**.

- **Idéia**:
  - Busca-se um plano desejado em vez de uma situação desejada (meta-busca).
  - Parte-se de um plano inicial (parcial) e aplica-se as ações até chegar a um plano final (completo)

- **Plano Final:**
  - **Completo**: todas as pré-condições de todas as ações são alcançada por meio de alguma outra ação.
  - **Consistente**: não há contradições.



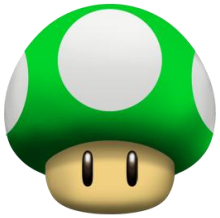
Planejamento de Ordem Parcial

- Na estratégia de **compromisso mínimo** a ordem e instanciações totais são decididas quando necessário.

- **Exemplo:**

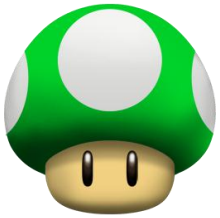
  - Para objetivo **Ter(Leite)**, a ação **Comprar(Produto, Loja)**, instancia-se somente item: **Comprar(Leite, Loja)**

  - Para o problema de colocar meias e sapatos: colocar cada meia antes do sapato, sem dizer por onde começar (esquerda ou direita)



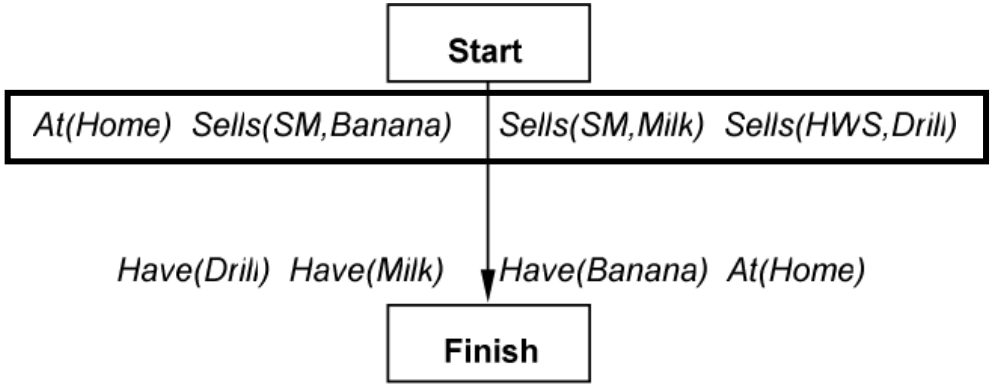
Planejamento de Ordem Parcial

- **Algoritmo de planejamento de ordem parcial:**
  - Identifica-se um passo com a pré-condição (sub-goal) não satisfeita.
  - Introduz-se um passo cujo efeito satisfaz a pré-condição.
  - Instancia-se variáveis e atualiza-se as ligações causais.
  - Verifica-se se há conflitos e corrigi-se o plano se for o caso.



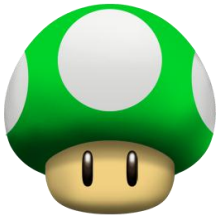
Exemplo

- Plano Inicial:

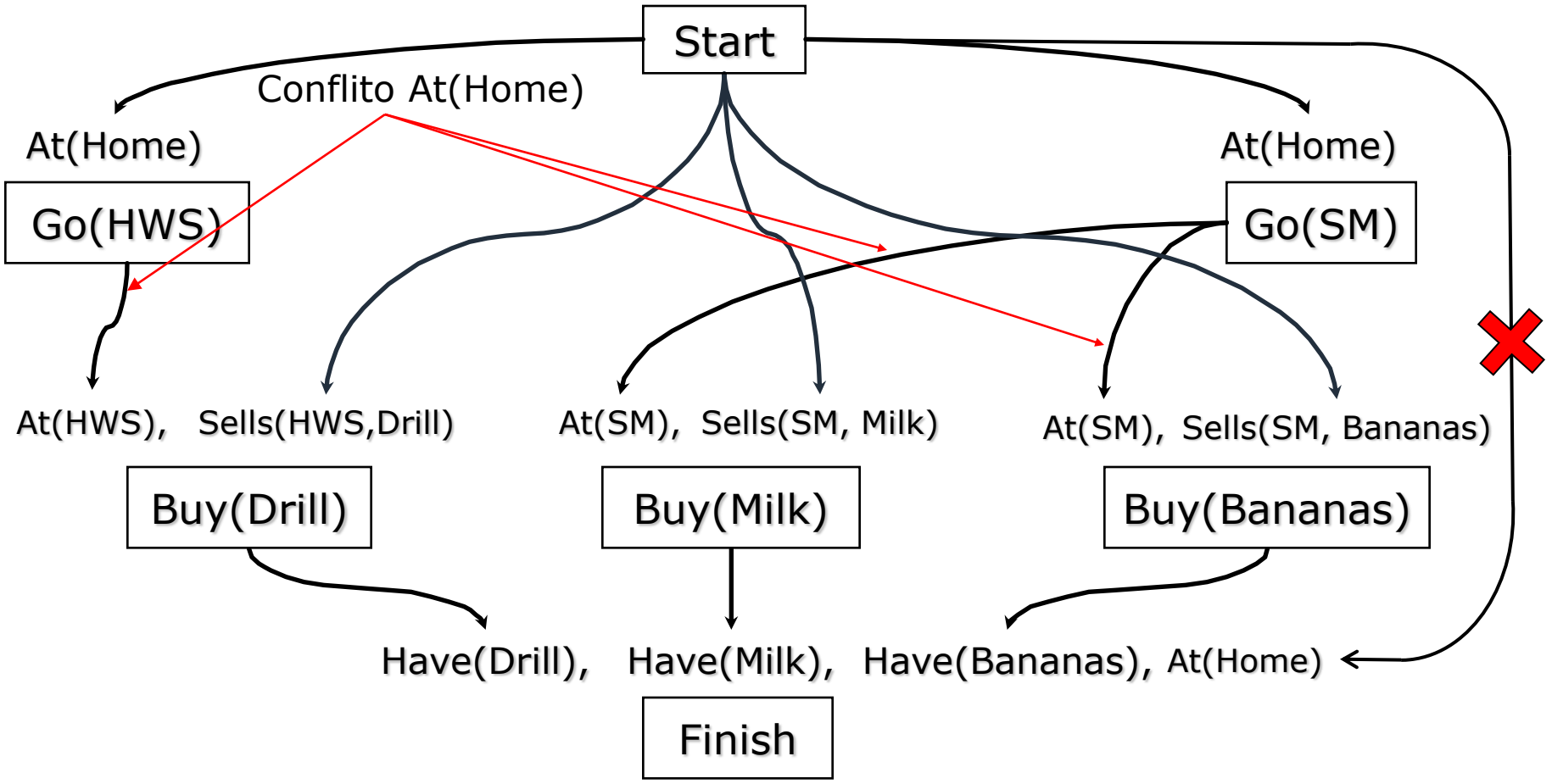


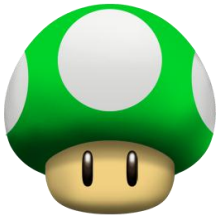
- Ações:

Op(ACTION: **Go**(there),
PRECOND: At(here),
EFFECT: At(there) $\wedge$ $\neg$ At(here))
Op(ACTION: **Buy**(x),
PRECOND: At(store) $\wedge$ Sells(store, x),
EFFECT: Have(x))



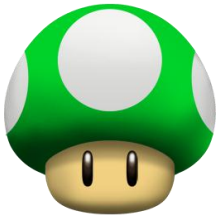
Exemplo


Start
Conflito At(Home)
At(Home)
Go(HWS)
At(Home)
Go(SM)
At(HWS), Sells(HWS,Drill)
At(SM), Sells(SM, Milk)
At(SM), Sells(SM, Bananas)
Buy(Drill)
Buy(Milk)
Buy(Bananas)
Have(Drill), Have(Milk), Have(Bananas), At(Home)
Finish



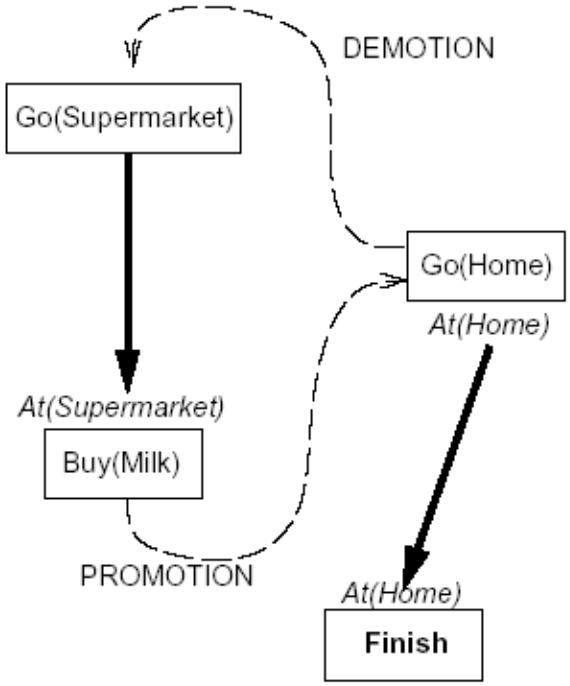
Conflito em Planejamento de Ordem Parcial

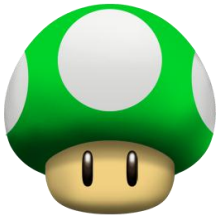
- Um **conflito** ocorre quando os efeitos de uma ação põem em risco as pré-condições de outra ação.
  - No caso anterior, os operadores Go(HWS) e Go(SM) apagam At(Home).



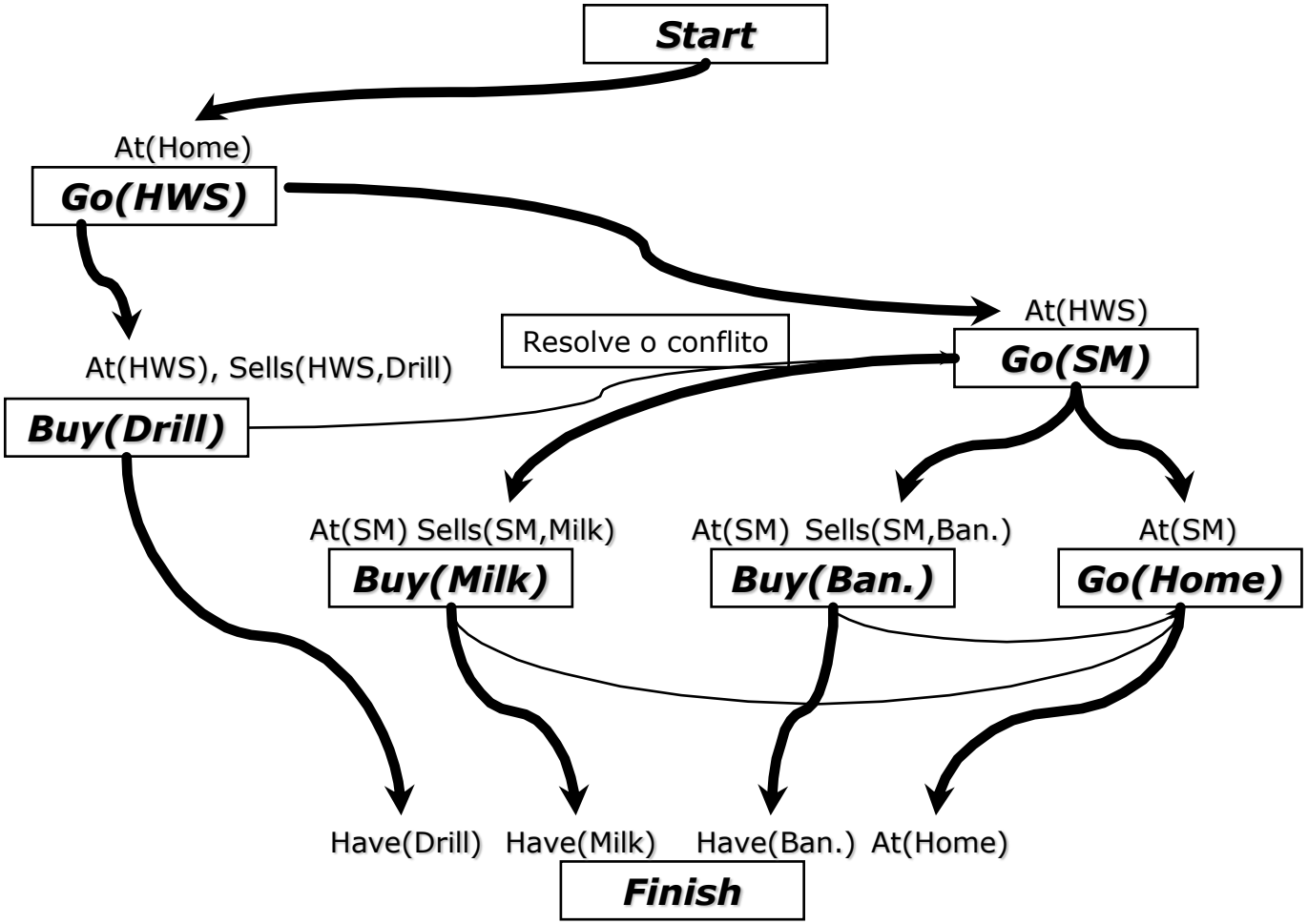
Solução de Conflitos

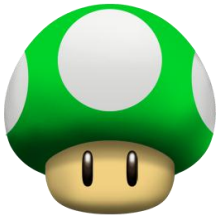
- Demotion e Promotion





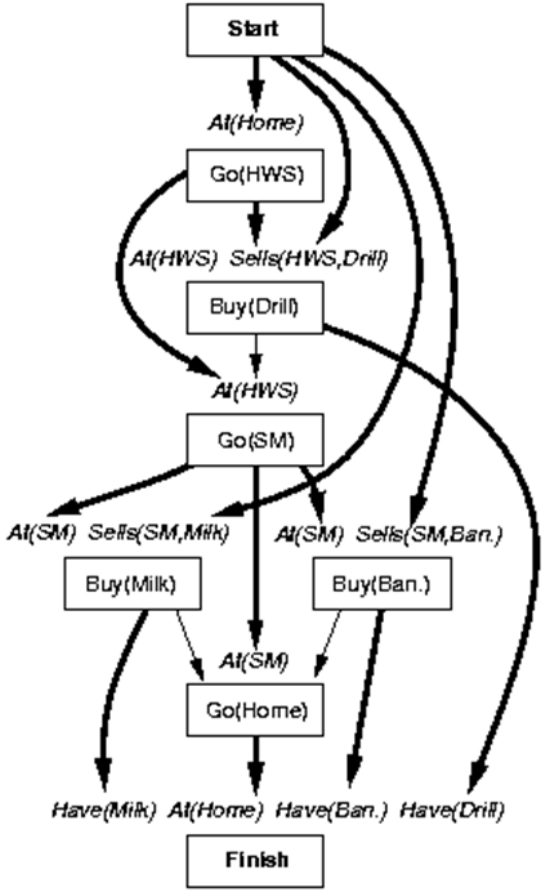
Exemplo

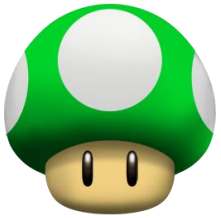

Start
At(Home)
Go(HWS)
At(HWS), Sells(HWS,Drill)
Buy(Drill)
Resolve o conflito
At(HWS)
Go(SM)
At(SM) Sells(SM,Milk)
Buy(Milk)
At(SM) Sells(SM,Ban.)
Buy(Ban.)
At(SM)
Go(Home)
Have(Drill)
Have(Milk)
Have(Ban.)
At(Home)
Finish



Exemplo

- Plano de Ordem Parcial





Aplicações de Planejamento

- Qualquer problema que necessite de **passos/ações** para chegar a um determinado **objetivo**.
- Exemplos:
  - Robôs que realizam tarefas.
  - Personagens de jogos direcionados a objetivos.
  - Geração de histórias para storytelling interativo.