

**Data Science
Academy**

www.datascienceacademy.com.br

Introdução à Inteligência Artificial

**Grafos de Planejamento e Algoritmo
Graphplan**

Todas as heurísticas que sugerimos podem sofrer de imprecisão. Uma estrutura de dados especial chamada grafo de planejamento pode ser usada para fornecer melhores estimativas de heurísticas. Essas heurísticas podem ser aplicadas a qualquer das técnicas de busca que vimos até agora. Como alternativa, podemos buscar uma solução sobre o espaço formado pelo grafo de planejamento utilizando um algoritmo chamado **Graphplan**. A pergunta de um problema de planejamento é se podemos alcançar um estado objetivo a partir do estado inicial. Suponha que seja dada uma árvore de todas as ações possíveis a partir do estado inicial para os estados sucessores, e seus sucessores, e assim por diante. Se indexarmos essa árvore adequadamente, poderemos responder à pergunta de planejamento sobre se “podemos atingir o estado G do estado S0” imediatamente, olhando para essa árvore. Naturalmente, a árvore é de tamanho exponencial, por isso essa abordagem é impraticável. Um grafo de planejamento é uma aproximação de tamanho polinomial dessa árvore que pode ser construída rapidamente. O grafo de planejamento definitivamente não pode responder se G é acessível de S0, mas pode estimar quantos passos levará para chegar a G. A estimativa é sempre correta quando ela relata que o objetivo não é alcançável e nunca superestima o número de passos, por isso é uma heurística admissível. Um grafo de planejamento é um grafo direcionado organizado em níveis: primeiro, o nível S0 para o estado inicial, que consiste de nós que representam cada proposição que é verdadeiro em S0; depois, o nível A0, consistindo de nós para cada ação instanciada que poderia ser aplicável em S0; em seguida, alterna níveis Si seguidos por Ai até chegar a uma condição de término.

Simplificando, Si contém todos os literais que poderiam ser verdadeiros no tempo i, dependendo das ações executadas nos passos de tempo anteriores. Se for possível P ou $\neg P$ ser verdade, ambos serão representados em Si. Também a grosso modo, Ai contém todas as ações que poderiam ter suas precondições satisfeitas no tempo i. Dizemos a grosso modo porque o grafo de planejamento registra apenas um subconjunto restrito de interações negativas possíveis entre ações e, portanto, um literal poderia aparecer no nível Sj quando, na verdade, poderia não ser verdadeiro, até num nível posterior, ou nunca aparecer (um literal nunca vai aparecer muito tarde). Apesar do possível erro, erro o nível j em que um literal aparece a princípio é uma boa estimativa do quão difícil é atingir o literal a partir do estado inicial.

Os grafos de planejamento funcionam apenas para problemas de planejamento proposicional — aqueles sem variáveis. Como já mencionamos, é simples proposicionalizar um conjunto de esquemas de ações.

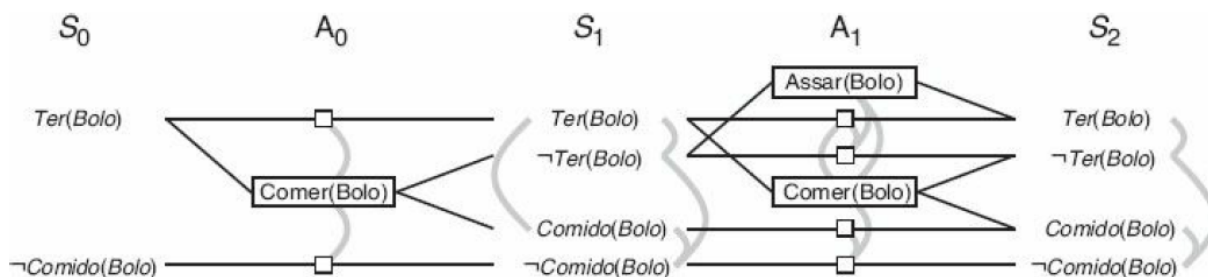
Apesar do aumento no tamanho da descrição do problema, os grafos de planejamento têm mostrado ser ferramentas eficazes para resolver problemas de planejamento difícil. A abaixo mostra um problema de planejamento simples, e a figura seguinte mostra seu grafo de planejamento. Cada ação no nível Ai está conectada a suas precondições em Si e seus efeitos em Si+1. Assim, um literal aparece porque uma ação o causou, mas também queremos dizer que um literal pode persistir se nenhuma ação negá-lo. Isso é representado pela ação de persistência. O nível A0 (primeira ação) na figura abaixo mostra uma ação “real”, Comer(Bolo), juntamente com duas ações de persistência desenhadas como pequenas caixas quadradas.

```

Início(Ter(Bolo))
Objetivo(Ter(Bolo) ∧ Comido(Bolo))
Ação(Comer(Bolo))
  PRECOND: Ter(Bolo)
  EFEITO: ¬Ter(Bolo) ∧ Comido(Bolo))
Ação(Assar(Bolo))
  PRECOND: ¬Ter(Bolo)
  EFEITO: Ter(Bolo))

```

O grafo de planejamento para o problema de “ter bolo e comer bolo” até o nível S2. Os retângulos indicam ações (pequenos quadrados indicam ações de persistência) e linhas retas indicam precondições e efeitos. As ligações de exclusão mútua (ou mutex) estão representados por linhas curvas em cor cinza. Nem todos as ligações de exclusão mútua são mostradas porque o grafo seria muito confuso. Em geral, se dois literais são mutex em S_i , as ações de persistência para aqueles literais serão mutex em A_i e não precisamos indicar essa ligação de mutex.



O nível A0 contém todas as ações que poderiam acontecer no estado S0, mas, quase tão importante quanto isso, ele registra conflitos entre ações que impediriam que essas ações ocorressem juntas. As linhas em cor cinza indicam ligações de exclusão mútua (ou mutex). Por exemplo, Comer(Bolo) é mutuamente exclusiva com a persistência de Ter(Bolo) ou ¬Comido(Bolo). Veremos em breve como as ligações de exclusão mútua são calculados.

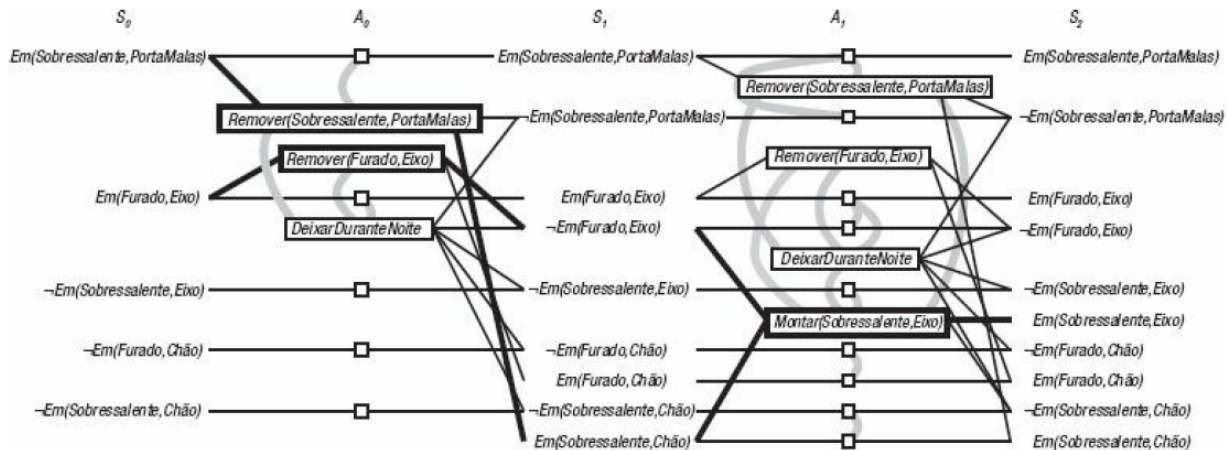
Uma vez construído, um grafo de planejamento é uma rica fonte de informações sobre o problema. Em primeiro lugar, se qualquer objetivo literal não aparece no nível final do grafo, o problema é insolúvel. Em segundo lugar, podemos estimar o custo de alcançar qualquer literal objetivo g_i do estado s como sendo o nível no qual g_i aparece pela primeira vez no grafo de planejamento construído do estado inicial s . Chamamos esse custo de custo de nível de g_i . Na figura anterior, Ter(Bolo) tem custo de nível 0 e Comido(Bolo) tem custo de nível 1. É fácil mostrar que essas estimativas são admissíveis para os objetivos individuais. Porém, talvez a estimativa não seja sempre exata porque os grafos de planejamento permitem várias ações em cada nível, enquanto a heurística leva em conta apenas o nível, e não o número de ações. Por essa razão, é comum se utilizar um grafo de planejamento serial para calcular heurísticas. Um grafo serial insiste em que apenas uma ação pode realmente acontecer em qualquer passo de

tempo dado; isso é feito adicionando-se ligações de exclusão mútua entre cada par de ações de não persistência. Os custos de nível extraídos de grafos seriais são muitas vezes estimativas bastante razoáveis dos custos reais.

Veja agora como extrair um plano diretamente do grafo de planejamento, em vez de apenas utilizar o grafo para fornecer uma heurística. O algoritmo GRAPHPLAN na figura abaixo, adiciona repetidamente um nível a um grafo de planejamento com EXPANDIR-GRAFO. Uma vez que todos os objetivos aparecem como sendo de não exclusão mútua no grafo, o GRAPHPLAN chama EXTRAIR-SOLUÇÃO para buscar um plano que resolva o problema. Se isso falhar, expande outro nível e tenta de novo, terminando em falha quando não houver razão para continuar.

```
função GRAPHPLAN(problema) retorna solução ou falha
  grafo ← GRAFO-DE-PLANEJAMENTO-INITIAL(problema)
  objetivos ← CONJUNÇÃO(problema.OBJETIVO)
  nãobons ← uma tabela hash vazia
  para tl = 0 até ∞ faça
    se objetivos são todos de não exclusão mútua em  $S_t$  do grafo então faça
      solução ← EXTRAIR-SOLUÇÃO(grafo, objetivos, NUMNÍVEIS(grafo), nãobons)
      se solução ≠ falha então retornar solução
    se grafo e nãobons estiverem ambos nivelados então retornar falha
    grafo ← EXPANDIR-GRAFO(grafo, problema)
```

Agora, vamos acompanhar a operação de GRAPHPLAN no problema do pneu sobressalente que vimos anteriormente. O grafo inteiro é mostrado na figura abaixo. A primeira linha do GRAPHPLAN inicializa o grafo de planejamento como um grafo de um único nível (S_0) representando o estado inicial. As proposições positivas do estado inicial da descrição do problema são mostradas, assim como as proposições negativas relevantes. Os literais positivos que não mudam não são mostrados (como Pneu(Sobressalente)) e os literais negativos irrelevantes. O literal de objetivo Em(Sobressalente, Eixo) não está presente em S_0 ; então não precisamos da chamada a EXTRAIR-SOLUÇÃO — estamos certos de que ainda não existe nenhuma solução. Em vez disso, EXPANDIR-GRAFO adiciona em A_0 as três ações cujas precondições existem no nível S_0 (isto é, todas as ações exceto Colocar(Sobressalente, Eixo)), junto com ações de persistência para todos os literais em S_0 . Os efeitos das ações são adicionados ao nível S_1 . Depois disso, EXPANDIR GRAFO procura por relações de exclusão mútua e as acrescenta ao grafo.



O grafo de planejamento para o problema de pneu sobressalente, depois da expansão até o nível S2. As ligações de exclusão mútua são mostradas como linhas em cor cinza. Nem todos os mutex são mostrados porque o grafo ficaria muito confuso se mostrássemos todos eles. A solução é indicada por linhas e contornos em negrito.

Em(Sobressalente, Eixo) ainda não está presente em S1, portanto mais uma vez não chamamos EXTRAIR-SOLUÇÃO. Chamamos EXPANDIR-GRAFO novamente, adicionando A1 e S1 resultando no grafo de planejamento. Agora, que temos o complemento total de ações, vale a pena examinar alguns dos exemplos de relações de exclusão mútua e suas causas:

- Efeitos inconsistentes: Remover(Sobressalente, PortaMalas) é mutex com DeixarDuranteNoite porque uma tem o efeito Em(Sobressalente, Chão) e a outra tem sua negação.
- Interferência: Remover(Furado, Eixo) é mutex com DeixarDuranteNoite porque uma ação tem a precondição Em(Furado, Eixo) e a outra tem sua negação como efeito.
- Necessidades concorrentes: Colocar(Sobressalente, Eixo) é mutex com Remover(Furado, Eixo), porque uma tem Em(Furado, Eixo) como precondição e a outra tem sua negação.
- Suporte inconsistente: Em(Sobressalente, Eixo) é mutex com Em(Furado, Eixo) em S2 porque a única maneira de alcançar Em(Sobressalente, Eixo) é através de Colocar(Sobressalente, Eixo), e essa ação é mutex com a ação de persistência que é o único modo de alcançar Em(Furado, Eixo). Desse modo, as relações mutex detectam o conflito imediato que surge a partir da tentativa de colocar dois objetos no mesmo lugar ao mesmo tempo.

Dessa vez, quando voltamos ao início da repetição, todos os literais do objetivo estão presentes em S2, e nenhum deles é mutex com qualquer outro. Isso significa que poderia existir uma solução, e EXTRAIR-SOLUÇÃO tentará descobri-la. Podemos formular, EXTRAIR-SOLUÇÃO como um problema de satisfação de restrição booleano (CSP) onde as variáveis são as ações em cada nível, os valores para cada variável estão dentro ou fora do plano, as restrições são as exclusões mútuas e há necessidade de satisfazer cada objetivo e precondição.



Referências:

Livro: Inteligência Artificial

Autor: Peter Norvig