Trabalho de Formatura Supervisionado
Planejamento Probabilístico

Helton Massato Kishi
Supervisora: Leliane Nunes de Barros

23 de julho de 2007

1 Introdução

Em Inteligência Artificial, Processos de Decisão Markovianos (*Markov Decision Processes* - MDPs) são utilizados para resolver problemas de planejamento com incertezas. A tarefa de planejamento pode ser definida por: dado um conjunto de ações, um estado inicial e um conjunto de estados meta, encontrar uma seqüencia de ações que leve um agente do estado inicial a um dos estados meta. Planejamento sob incerteza é uma extensão do problema de planejamento clássico, em que as ações podem ter efeitos incertos.

O estado atual em que o agente se encontra e a escolha da ação a ser tomada naquele momento determinam a distribuição de probabilidade do próximo estado a ser visitado. É necessário também definir estados finais (estados meta) do sistema. Estes são os estados em que o sistema gostaria de chegar. O objetivo do planejador é criar uma *política*, que mapeia estados em ações, que maximize as chances de se chegar nos estados finais a partir de um estado inicial.

1.1 Estados e transição de estados

O estado nada mais é do que uma descrição do sistema num dado momento. Um estado pode ser representado, por exemplo, através de um conjunto de cláusulas de primeira ordem (modelo *fatorado*), ou simplesmente enumerando-se todos os estados possíveis do sistema (modelo *enumerativo*).

A transição de estados é definida por uma matriz de probabilidade. Para cada estado e para cada ação, são definidas as probabilidades de se chegar aos outros estados. A soma das probabilidades de transição de um dado estado, executando-se uma dada ação, deve ser 1. Mais precisamente, seja S um espaço de estados e A um espaço de ações,

$$\forall s \in S, \forall a \in A, \sum_{s' \in S} P_a(s'|s) = 1$$

onde $P_a(s'|s)$ significa a probabilidade do agente chegar ao estado s' dado que seu estado atual é s e a ação executada foi a.

Note que a transição de estados do planejamento clássico (isto é, planejamento determinístico) também pode ser representada usando-se esta matriz. No

entanto, para cada estado e ação, apenas *um* dos estados tem valor 1 e os outros tem valor 0.

Podemos representar (graficamente) um MDP através de um grafo dirigido. Cada nó do grafo é um estado e de cada estado saem arestas para outros estados. Cada aresta tem um valor variando de 0 a 1, representando a probabilidade de sairmos do estado origem da aresta e chegarmos no estado destino da aresta. A soma de todas as arestas saindo de um mesmo nó deve ser sempre 1.

1.2 Trajetória e história do sistema

Os termos *trajetória* e *história* são utilizados para descrever o comportamento do sistema durante o momento de solucionar o problema. A história completa de um sistema é a seqüencia de estados, ações e observações geradas desde o estado 0 até um certo instante de tempo de interesse, podendo ser tanto finita, quanto infinita. Uma história pode ser representada por:

$$(< S_0, O_0, A_0 >, < S_1, O_1, A_1 >, \ldots, < S_t, O_t, A_t >)$$

No caso de sistemas totalmente observáveis, é necessário apenas os estados em que o sistema passou e as ações tomadas.

$$(< S_0, A_0 >, < S_1, A_1 >, \ldots, < S_{t-1}, A_{t-1} >, S_t)$$

1.3 Função Valor e Recompensa

Para que o agente consiga ver a qualidade das ações tomadas, é necessário uma função valor $V : H_s \rightarrow \mathbf{R}$. O agente prefere a história h ao invés de h' se $V(h) > V(h')$.

Esta função Valor pode ser definida da seguinte forma, usando outras duas funções primitivas que podem ser aplicadas a estados e ações. São elas:

- **Função recompensa: $R : S \rightarrow \mathbf{R}$ que indica a recompensa de estar no estado s. Nesse trabalho, as recompensas assumirá valores positivos.**
- **Função custo: $C : S \times A \rightarrow \mathbf{R}$ que indica o custo de aplicarmos a ação a no estado s. Nesse trabalho, custo é sempre menor ou igual a zero.**

A função valor é calculada fazendo-se uma simples combinação linear de custo e recompensa. Se fizermos a soma de todos os custos e recompensas para todos os estados da história Hs, estaremos calculando uma função de valor *time-separable*. No entanto, pode existir a necessidade de que os custos e recompensas sejam variáveis ao longo do tempo de execução do sistema. Porém, isto pode ser contornado criando-se mais estados, um para cada período de tempo.

1.4 Horizonte e critério de sucesso

Para o cálculo da função valor, é necessário explicitar um tempo T de duração da simulação, chamado de *horizonte*.

$$V(h) = \sum_{t=0}^{T-1} \{R(s_t) - C(s_t, a_t)\} + R(s_T).$$

Podemos também definir um problema com horizonte infinito:

$$V(h) = \sum_{t=0}^{\infty} (\gamma^t.(R(s_t) - C(s_t, a_t)))$$

onde γ é a taxa de desconto ($0 \leq \gamma < 1$). Quanto menor o γ, menor será a importância dos acontecimentos longe do início.

Outro jeito de lidar com o problema de horizontes infinitos é avaliar a trajetória baseada na média de recompensa por estado, chamado de *ganho*.

1.5 Política

Política é um mapeamento de histórias observáveis para ações, ou seja,

$$\pi : H_o \rightarrow A.$$

O agente executa a ação:

$$a_t = \pi(<< o_0, a_0 >, \ldots, < o_{t-1}, a_{t-1} >, o_t >)$$

No caso de MDP totalmente observável com função valor *time-separable*, a ação ótima pode ser computada usando apenas a informação do estado atual e tempo, ou seja, um política π é dada por:

$$\pi : S \times T \rightarrow A$$

1.6 O problema de planejamento probabilístico

Para se definir um problema de planejamento probabilístico, são necessários os seguintes itens de entrada (dados do problema):

1. S é um espaço de estados finito.
2. $s_0 \in S$ é o estado inicial.
3. $G \subset S$ é um conjunto de estados finais.
4. A é o conjunto de ações possíveis no sistema.
5. Matriz de probabilidade de tamanho $|S| \times |S| \times |A|$ definida por $P_a(s'|s)$, isto é, a probabilidade do sistema ir pra o estado s', dado que o estado atual é s e a ação tomada foi a. Esta função deve estar definida para todo $a \in A$ e $s, s' \in S$.
6. Uma função custo de ações;
7. Uma função recompensa de estados;
8. Horizonte T.

O planejador probabilístico deve devolver uma política ótima que maximize a recompensa obtida pelos estados visitados e minimize o custo das ações executadas pelo agente. Para alguns problemas em que a função recompensa não é dada, o objetivo é encontrar uma política que apenas minimize o custo.

1.7 Programação dinâmica: um método para resolver um MDP

A idéia do algoritmo é inicializar um vetor $V(s)$ com valores arbitrários para estados não finais e 0 para estados finais. A cada iteração, usa-se a equação:

$$V(s) = \min_{a \in A(s)} \{c(a,s) + \sum_{s' \in S} P_a(s'|s)V(s')\}$$

para a atualização dos valores de $V(s)$. Assim, com o tempo, $V(s)$ tende a convergir para os valores ótimos. A cada passo, o planejador olha para o estado atual s, e escolhe a ação que corresponde ao menor valor de $V(s)$. Podemos definir um *residual* $\epsilon(s)$, que é a diferença entre os valores entre uma iteração e outra. Se esta diferença for muito pequena, o algoritmo termina.

Esse algoritmo, chamado de *iteração por valor*, resolve a equação acima por programação dinâmica. A equação acima é chamada de *equação de Bellman*.

2 Objetivo

O objetivo desse trabalho é realizar um estudo das diferentes técnicas de solução para problemas de planejamento probabilístico, usando Processos Markovianos de Decisão (MDPs). Para isso, será necessário entender como modelar problemas de planejamento como um MDP e como utilizar técnicas avançadas de Inteligência Artificial para resolver o MDP resultante, de modo mais eficiente do que o algoritmo de iteração por valor. Esse estudo será baseado em trabalhos recentes da área, publicados em periódicos e anais de congressos. Serão implementados dois algoritmos clássicos para resolver problemas de planejamento como MDPs, a saber: *iteração por valor* e *iteração por política*. Além disso, será implementado um algoritmo mais eficiente para planejamento com MDPs que utiliza técnicas avançadas de busca heurística da Inteligência Artificial.

3 Atividades e cronograma

- [**Junho - Julho**] Um estudo sobre as diferentes linhas de pesquisa na área de Planejamento em Inteligência Artificial .
- [**Junho - Julho**] Levantamento bibliográfico na área de planejamento sob incerteza.
- [**Agosto - Setembro**] Estudo e implementação dos algoritmos clássicos de resolução de MDPs.
- [**Outubro**] Estudo e implementação de algoritmos mais eficientes para planejamento usando MDPs.
- [**Setembro - Outubro**] Seleção de domínios de planejamento como casos de teste.
- [**Outubro - Novembro**] Avaliação de desempenho dos algoritmos implementados.
- [**Outubro**] Elaboração do poster.

- **[Setembro - Novembro]** Escrita da monografia.

4 Estrutura da monografia

```
1. Introdução
  1.1 Motivação
  1.2 Definições e teoria
  1.3 Objetivos do trabalho
2. Planejamento em Inteligência Artificial
  2.1 Planejamento Determinístico
  2.2 Planejamento sob incerteza
    2.2.1 Planejamento com efeitos não-determinísticos
    2.2.2 Planejamento com efeitos probabilísticos
3. Planejamento probabilístico usando MDPs
4. Algoritmos para MDPs e implementações
5. Uso de heurísticas para resolver MDPs para planejamento
    5.1 O algoritmo RTDP
    5.2 O algoritmo LRTDP
6  Análise experimental
7. Conclusão e trabalhos futuros
8. Parte subjetiva sobre a realização do trabalho
   8.1 Desafios e frustrações encontrados
   8.2 Lista das disciplinas cursadas no BCC mais relevantes para o trabalho
9. Bibliografia
```

5 Bibliografia

- Boutilier, Craig *Decision Theoretic Planning: Structural Assumptions and Computational Leverage*
- Bonet, Blai *Labeled RTDP: Improving the Convergence of Real-Time Dynamic Programming*