Um modelo unificado para planejamento sob incerteza

Felipe Werndl Trevizan

DISSERTAÇÃO APRESENTADA
AO
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DA
UNIVERSIDADE DE SÃO PAULO
PARA
OBTENÇÃO DO TÍTULO DE MESTRE
EM
CIÊNCIAS

Área de Concentração: Ciência da Computação
Orientador: Profa. Dra. Leliane Nunes de Barros

Durante o desenvolvimento deste trabalho, o aluno recebeu apoio financeiro do CNPq (processo 131403/05-2)

— São Paulo, maio de 2006. —

Um modelo unificado para planejamento sob incerteza

Este exemplar corresponde à redação final da dissertação devidamente corrigida e defendida por Felipe Werndl Trevizan e aprovada pela banca julgadora.

São Paulo, maio de 2006.

Banca examinadora:

Profa. Dra. Leliane Nunes de Barros (presidente)	IME-USP
Prof. Dr. Fabio Gagliardi Cozman	POLI-USP
Prof. Dr. Ronaldo Fumio Hashimoto	IME-USP

Resumo

Dois modelos principais de planejamento em inteligência artificial são os usados, respectivamente, em planejamento probabilístico (MDPs e suas generalizações) e em planejamento não-determinístico (baseado em *model checking*). Nessa dissertação será: (1) exibido que planejamento probabilístico e não-determinístico são extremos de um rico contínuo de problemas capaz de lidar simultaneamente com risco e incerteza (Knightiana); (2) obtido um modelo para unificar esses dois tipos de problemas usando MDPs imprecisos; (3) derivado uma versão simplificada do princípio ótimo de Bellman para esse novo modelo; (4) exibido como adaptar e analisar algoritmos do estado-da-arte, como (L)RTDP e LDFS, nesse modelo unificado. Também será discutido exemplos e relações entre modelos já propostos para planejamento sob incerteza e o modelo proposto.

Abstract

Two noteworthy models of planning in AI are probabilistic planning (based on MDPs and its generalizations) and nondeterministic planning (mainly based on model checking). In this paper we: (1) show that probabilistic and nondeterministic planning are extremes of a rich continuum of problems that deal simultaneously with risk and (Knightian) uncertainty; (2) obtain a unifying model for these problems using imprecise MDPs; (3) derive a simplified Bellman's principle of optimality for our model; and (4) show how to adapt and analyze state-of-art algorithms such as (L)RTDP and LDFS in this unifying setup. We discuss examples and connections to various proposals for planning under (general) uncertainty.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Siglas, Símbolos e Funções

$\mathbf{F}(s,a)$	função de transição de estados para MDPSTs, *p. 31*.
γ	fator de desconto para MDPs de horizonte infinito, *p. 18*.
$\mathcal{E}$	conjunto de eventos exógenos, *p. 11*.
$\mathcal{H}$	espaço de históricos, *p. 18*.
$\mathcal{O}$	conjunto de observações, *p. 12*.
Ω	espaço de possibilidades, *p. 5*.
ω	estado da natureza ($\omega \in \Omega$), *p. 5*.
Π	espaço de políticas, *p. 19*.
π	política, *p. 14*.
π^*	política ótima, *p. 19*.
Σ	sistema de transição de estados para simular um ambiente real, *p. 12*.
$C(s,a)$	custo de aplicar a ação a no estado s, *p. 15*.
$E[\pi]$	custo esperado da política π, *p. 18*.
$F(s,a)$	função de transição de estados, *p. 15*.
k	conjunto de estados alcançáveis, *p. 31*.
O	função de observação, *p. 12*.
$P_0(\cdot)$	medida de probabilidade para o estado inicial de um MDP, *p. 17*.
$P_{st}(\cdot)$	medida de probabilidade induzida pelo critério minimax dos MDPSTs, *p. 49*.
s_0	estado inicial, *p. 11*.
S_π	domínio da política π, *p. 16*.
S_G	conjunto de estados meta, *p. 11*.
$S_\pi^{s_0}$	espaço gerado por s_0 e π, *p. 23*.
T	função geral de transição de estados, *p. 11*.
$V(h)$	custo do histórico h, *p. 18*.
V^π	função valor para políticas, *p. 19*.
MBE1 – MBE6	modelo básico de estados para planejamento, *p. 15*.
MDPIP1 – MDPIP5	processo de decisão markoviano com probabilidades imprecisas, *p. 33*.
MDPST1 – MDPST5	processo markoviano de decisão com transição valorada por conjunto, *p. 31*.
MDP1 – MDP5	processo markoviano de decisão, *p. 17*.
MGP1 – MGP8	modelo geral de planejamento, *p. 11*.
PC1 – PC4	axiomas de probabilidade condicional, *p. 7*.
PC4'	axioma alternativo à PC4, *p. 7*.
PTD1 – PTD3	modelo de um problema de decisão, *p. 9*.
PU1 – PU3	axiomas de probabilidade incondicional, *p. 6*.
SPST1 – SPST6	caminho mínimo com transição valorada por conjuntos, *p. 37*.
SSP1 – SSP6	caminho estocástico mínimo, *p. 19*.
$\mathcal{K}$	conjunto credal, *p. 6*.
$\mathcal{A}$	espaço de ações, *p. 11*.
$\mathcal{S}$	espaço de estados, *p. 11*.
GPS	solucionador geral de problemas, *p. 1*.

IA	inteligência artificial, *p. 1*.
LRTDP	Labeled Real Time Dynamic Programming, *p. 23*.
MDP	processo markoviano de decisão, *p. 17*.
MDPIP	processo de decisão markoviano com probabilidades imprecisas, *p. 33*.
MDPST	processo markoviano de decisão com transição valorada por conjunto, *p. 31*.
PTD	problema de tomada de decisão, *p. 9*.
SPST	caminho mínimo com transição valorada por conjuntos, *p. 37*.
SSP	problema do caminho estocástico mínimo, *p. 19*.

Introdução

Um homem que não planeja seus passos com antecedência
encontrará problemas logo em sua porta.

— Confúcio, 551 a.C.–479 a.C.
(filósofo chinês)

O desenvolvimento de um *solucionador geral de problemas* (*general problems solver* — GPS) tem sido um dos principais objetivos da área de inteligência artificial IA [Geffner, 2002]. Um GPS é um programa que recebe como entrada uma descrição de alto-nível de um problema e automaticamente computa a sua solução [Newell e Simon, 1963]. Existem duas motivações para o desenvolvimento de tais programas. A primeira motivação é cognitiva, i.e., seres humanos são solucionadores gerais de problemas, assim, reproduzir ou simplesmente emular tal característica é um grande desafio para a inteligência artificial. A segunda motivação é técnica: modelar problemas em alto-nível de abstração para um GPS resolver é mais simples do que desenvolver programas específicos para cada problema. Dessa forma, um GPS pode ser uma ferramenta muito útil na prática.

Para permitir a um projetista modelar problemas em alto-nível de abstração é preciso fornecer uma linguagem geral para descrever tais problemas, bem como algoritmos gerais para resolvê-los. Enquanto as soluções obtidas por tais sistemas podem não ser tão boas ou tão rápidas quanto aquelas geradas por algoritmos dedicados (*ad hoc*), o uso de um GPS pode ser justificado se o seu desempenho for similar ao de uma solução dedicada, ou ainda, se a implementação de uma solução dedicada for muito dispendiosa.

Para desenvolver um GPS, primeiro é necessário definir claramente o seu escopo de atuação, caso contrário não será possível projetar uma linguagem, nem algoritmos para ele. Uma maneira possível para definir tal escopo é através da escolha de um modelo matemático, cujas instâncias serão resolvidas pelo GPS. Por exemplo, um modelo determinístico de transição de estados, i.e., um modelo no qual uma ação mapeia deterministicamente um estado em outro, define o seguinte escopo de atuação de um GPS: encontrar uma *seqüência ordenada de ações* (pois o efeito de cada ação é determinístico) que satisfaça as restrições fornecidas pelo problema recebido como entrada. Ao introduzir outras características ao escopo do GPS, como por exemplo incerteza nos efeitos das ações ou ações com duração de tempo, outros modelos matemáticos devem ser usados, aumentando a expressividade da linguagem para descrever problemas, bem como a complexidade dos algoritmos de solução.

Planejamento em inteligência artificial é a área que estuda o desenvolvimento de solucionadores gerais de problemas para determinadas classes de modelos matemáticos de ações e estados. Além de definirem o escopo do GPS, nesse caso chamado de planejador, que será desenvolvido, esses modelos também determinam: o tipo de problemas que o planejador deverá resolver; a forma das soluções e as características desejadas dessas soluções (ótimas, sub-ótimas, etc). Dessa maneira, é possível ver planejamento em IA como o estudo de representações convenientes (classes de modelos matemáticos) de problemas e de algoritmos eficientes capazes de solucionar, de forma automática, qualquer instância do modelo matemático escolhido.

Em planejamento, a forma com que um modelo evolui ao aplicar uma ação, chamada de *dinâmica das ações*, pode ser: determinística, não-determinística ou probabilística. Enquanto os modelos determinísticos não modelam incerteza no efeito das ações, os modelos não-determinísticos e probabilísticos modelam diferentes formas de incerteza para o efeito das ações. Apesar do modelo determinístico ser um caso especial dos

outros dois modelos, não existe uma relação direta entre os modelos não-determinísticos e probabilísticos. A independência entre essas duas dinâmicas de ações sugere a existência de problemas que envolvam ao mesmo tempo ações probabilísticas e não-determinísticas. Nesse trabalho, será proposto um modelo matemático para planejamento cuja dinâmica das ações suporta diferentes formas de incerteza, o que resulta em um modelo no qual é possível expressar tanto os problemas anteriores quanto uma nova classe de problemas, chamada de problemas de *planejamento sob incerteza*.

Motivação

Uma das principais abordagens da área de planejamento em IA é fornecida por Bonet e Geffner [2006; 2001c]. Em seus trabalhos, as diferentes formas de planejamento são descritas através de características comuns entre si, formulando linguagens, modelos e algoritmos para casos gerais de planejamento. Como será discutido na Seção 2.3, esse ponto de vista tem sido eficiente na unificação de diferentes linhas de pesquisa dentro de planejamento, do clássico ao probabilístico, incluindo variações de planejamento não-determinístico. Tal formulação unificada é benéfica tanto para alcançar os objetivos da IA quanto para outras áreas de pesquisas, como por exemplo, a área de pesquisa operacional.

Uma das limitações da formulação de Bonet e Geffner é tratar ações probabilísticas e ações não-determinísticas como duas características mutuamente exclusivas. Tais propriedades se baseiam em diferentes hipóteses feitas sobre o conhecimento da dinâmica do ambiente modelado, respectivamente, sempre traduzida em termos de probabilidades ou nunca traduzida em termos de probabilidades. Essas hipóteses também induzem algumas propriedades da solução: na dinâmica probabilística a solução é usualmente baseada no valor esperado da função de utilidade [Boutilier *et al.*, 1999], enquanto a dinâmica não-determinística obriga o planejador a oferecer garantias sobre atingir determinados estados independentemente do não-determinismo (pior caso).

Dessa forma, pesquisas em planejamento probabilístico e não-determinístico possuem pouca interação entre elas. De certa maneira, essa falta de interação é um reflexo do contraste geral na área de teoria da decisão entre a linha Bayesiana e a linha *minimax*: enquanto a primeira está associada ao valor esperado de uma função, a segunda ao comportamento no pior caso dessa mesma função. Porém, a área da teoria de decisão possui muitas outras abordagens, em especial há abordagens capazes de lidar tanto com o valor esperado e com o comportamento no pior caso de funções quanto com qualquer combinação dessas duas abordagens. Assim, é possível modelar problemas onde alguns eventos possuem probabilidades associadas enquanto outros eventos ocorrem de forma não-determinística.

Objetivo

A proposta desse trabalho é desenvolver um modelo matemático de estados e ações capaz de representar de maneira unificada problemas de *planejamento sob incerteza*. Nesse modelo, planejamento probabilístico e planejamento não-determinístico são vistos como casos especiais, o que revela uma nova gama de problemas que ainda não foram abordados pela comunidade de planejamento em IA.[1]

Outro objetivo desse trabalho é demonstrar que esses novos problemas, envolvendo ações que são probabilísticas e não-determinísticas ao mesmo tempo, são *processos markovianos de decisão com probabilidades imprecisas* [Satia e Lave Jr, 1973; White III e Eldeib, 1994]. Também faz parte desse trabalho exibir como adaptar os algoritmos para resolver instâncias desse modelo, em especial os usado em pesquisa operacional, para problemas de *planejamento sob incerteza*.

Organização

Essa dissertação está organizada da seguinte maneira:

[1] Planejamento determinístico também está incluído nesse modelo porque ele pode ser visto como um caso degenerado tanto de planejamento probabilístico quanto não-determinístico.

Capítulo 1 Uma breve revisão sobre os conceitos básicos de probabilidades e da teoria da decisão. Também é fornecida uma discussão sobre as diferentes manifestações de incerteza com o intuito de construir uma linguagem padrão para esse texto.

Capítulo 2 Uma descrição sucinta da área de planejamento através do ponto de vista unificado de Bonet e Geffner [2006; 2001c] e de Ghallab *et al.* [2004].

Capítulo 3 Apresentação do modelo para problemas de *planejamento sob incerteza*, bem como resultados e algoritmos para tal modelo.

Capítulo 4 Levantamento bibliográfico e resumo dos artigos da área de planejamento e pesquisa operacional que sejam relacionados a essa proposta.

Capítulo 5 Apresentação dos resultados obtidos através desse estudo, bem como dos pontos nos quais esse trabalho pode ser estendido.

Apêndice A Prova do principal resultado envolvendo o modelo proposto, o teorema 3. Esse teorema fornece o alicerce para a solução de instâncias desse modelo.

Na organização lógica do texto, os Capítulos 1 e 2 representam os pré-requisitos necessários para a compreensão desse trabalho. O Capítulo 3 contém o modelo proposto e os resultados obtidos, o Capítulo 4 faz uma revisão bibliográfica sobre trabalhos correlatos e por último, o Capítulo 5 apresenta as conclusões desse trabalho e os próximos passos a serem seguidos para estendê-lo.

Capítulo 1

Uma breve revisão sobre probabilidades e teoria da decisão

Quando não está em nosso poder determinar o que é verdadeiro, devemos seguir o que é mais provável.

— René Descartes, 1596–1650
(matemático e filósofo francês)

Nesse capítulo serão revistos alguns conceitos básicos de probabilidades [Cozman, 2005a], bem como será fornecida uma breve discussão sobre teoria da decisão [Cozman, 1997; Trevizan *et al.*, 2006]. A interpretação de probabilidades considerada nesse texto é subjetivista [Cheeseman, 1985]. Nessa abordagem, as probabilidades são uma medida da crença ou ignorância de quem as especificou devido sua insuficiência de conhecimento sobre fenômeno modelado.

1.1 Espaço de possibilidades, estados e eventos

O alicerce da teoria de probabilidades é o *espaço de possibilidades*, também chamado de espaço amostral. O espaço de possibilidades, termo que será adotado nesse texto, é denotado por Ω e representa o conjunto (não-vazio) de todas as possíveis saídas do fenômeno que será modelado. Os elementos ω do conjunto Ω são chamados de estado da natureza e são considerados mutuamente exclusivos.[1] Dessa forma, em qualquer momento o fenômeno modelado será representado por um, e apenas um, $\omega \in \Omega$. Nesse estudo será considerado que o espaço de possibilidades Ω é enumerável, finito ou infinito.

Outro conceito importante é o de *eventos*. Um evento é um subconjunto de Ω e será denotado por letras maiúsculas, por exemplo A, B, C, etc. Quando um evento A contém o estado da natureza observado após a ocorrência do fenômeno modelado, é dito que o evento A ocorreu. Para denotar todos os estados da natureza que não pertencem a um evento A, i.e. o complemento de A em relação à Ω, é adicionado a letra c sobrescrita ao nome do evento (A^c).

Dado dois eventos A e B, sua intersecção representa todos os estados da natureza que estão ao mesmo tempo em A e B e é denotada por $A \cap B$. A união desses dois eventos, denotada por $A \cup B$, representa os estados da natureza pertencentes a A, B ou $A \cap B$.

[1] Os elementos de Ω são usualmente chamados apenas de estados, mas para evitar conflito com o termo *estado* da área de planejamento, será adotado sua versão mais longa.

1.2 Variáveis aleatórias

Definido um espaço de possibilidades, é possível construir uma função que relacione os elementos de Ω com os números reais. Qualquer função X com essa característica, ou seja, $X : \Omega \to \mathbb{R}$, é chamada de variável aleatória ou simplesmente de variável. Note que, a função X mapeia de forma determinística todo $\omega \in \Omega$ à um número real, sendo que o termo aleatório se refere à *incerteza* associada ao fenômeno modelado.

Exemplo 1. *A idade de uma pessoa ω selecionada de uma população Ω é uma variável aleatória $X(\omega)$. Sobre a mesma população Ω é possível definir uma outra variável aleatória Y, em que $Y(\omega)$ representa o peso (arredondado para valores em quilogramas) de uma pessoa ω que pertence à população Ω.*

1.3 Medidas de probabilidade

Uma *medida de probabilidade* é uma função P que associa a cada evento A de Ω um peso $P(A)$, chamado de probabilidade. Como não é feita nenhuma suposição sobre algum evento ter ocorrido anteriormente, a função P também pode ser chamada de medida de probabilidade incondicional ou a priori. Entre as suposições feitas, a primeira é que o valor da probabilidade de um evento deve ser um número real entre zero e um (incluindo ambos). Atribuir probabilidade zero a um evento representa a impossibilidade dele ocorrer, enquanto associar o valor um está relacionado com certeza de que esse evento irá ocorrer.

Além disso é necessário a seguinte suposição: dado dois eventos A e B tais que sejam possível definir precisamente $P(A)$ e $P(B)$, se A e B forem eventos disjuntos então $P(A \cup B) = P(A) + P(B)$. Quando a probabilidade de todos os eventos de Ω estão precisamente especificadas, essas suposições podem ser resumidas nos seguinte axiomas:

PI1 Para qualquer evento A, $P(A) \geq 0$;

PI2 O espaço de possibilidades tem probabilidade um, $P(\Omega) = 1$;

PI3 Se dois eventos são disjuntos, i.e., $A \cap B = \emptyset$, então $P(A \cup B) = P(A) + P(B)$.

Através dos axiomas de probabilidade incondicional (PU1 – PU3) é possível calcular a probabilidade do complemento de um evento A por: $P(A^c) = 1 - P(A)$, pois $P(A) + P(A^c) = P(\Omega) = 1$. Outro resultado desses axiomas é: $P(A \cup B) = P(A) + P(B) - P(A \cap B)$ para quaisquer eventos A e B de Ω.

1.4 Conjuntos de medidas de probabilidade

Apesar de simples e intuitivo estabelecer uma probabilidade para cada evento, na prática, obter esse valor exato é muito difícil e em alguns casos impossível. Uma forma de transpor essa dificuldade é permitir que a probabilidade de cada evento seja definida através de restrições, que também são chamadas de afirmações (*assessments*) sobre o fenômeno modelado. A coleção dessas afirmações gera um conjunto de medidas de probabilidade, chamado de conjunto credal [Cozman, 2005b], que será denotado por $\mathcal{K}$.

Note que, todo conjunto credal construído com pelo menos duas afirmações conflitantes é vazio e será denominado inválido. Já os conjuntos credais que possuírem pelo menos uma medida de probabilidade, i.e. $|\mathcal{K}| > 0$, são chamados de válidos. Além da caracterização feita sobre a norma de $\mathcal{K}$, ainda é possível caracterizar todo conjunto credal válido como conexo ou não e aberto ou fechado.

Dado um conjunto credal válido, é possível inferir novas restrições sobre a probabilidade de outros eventos usando os axiomas de probabilidade incondicional e seus resultados. Essas novas restrições podem definir precisamente ou limitar a probabilidade de eventos, dependendo do conjunto inicial de afirmações e da capacidade computacional disponível.

Para não sobrecarregar a notação, a partir desse ponto do texto, todo conjunto credal válido será chamado apenas de conjunto credal. Quando necessário, será explicitado que o conjunto credal em questão é inválido.

1.5 Medidas de probabilidade condicional

Em muitas situações é interessante considerar a probabilidade de um evento A condicionado na ocorrência de outro evento B. A *probabilidade condicional*, representada por $P(A|B)$, expressa esse conceito. Formalmente, dado um espaço de possibilidades Ω, uma medida de probabilidade condicional é uma função $P(\cdot|\cdot)$ que mapeia pares de eventos em números reais entre 0 e 1 (ambos inclusos). O primeiro evento é chamado de evento condicionado, enquanto o segundo é chamado de evento condicionante. Note que o evento condicionante não pode ser vazio, pois o evento vazio é a representação da impossibilidade lógica, o que impede a especulação da probabilidade de qualquer outro evento dado sua ocorrência.[2]

Ao fixar um evento condicionante B não-vazio, a medida $P(\cdot|B)$ satisfaz os axiomas de probabilidade incondicional. Além disso, é possível adaptar PU2 para medidas de probabilidade condicionais, resultando em $P(B|B) = 1$, o que é bem intuitivo, pois é considerado que o evento B já ocorreu. Dessa forma, é possível construir os *axiomas de probabilidade condicional* (PC1 – PC4).

PC1 Para qualquer evento A e evento não-vazio B, $P(A|B) \geq 0$.

PC2 Para qualquer evento B não-vazio: $P(\Omega|B) = P(B|B) = 1$

PC3 Se dois eventos são disjuntos, então para qualquer evento não-vazio C, vale $P(A \cup B|C) = P(A|C) + P(B|C)$.

PC4 Dado três eventos, A, B, C tais que B e $B \cap C$ são não-vazios, vale $P(A \cap B|C) = P(A|B \cap C)P(B|C)$.

O axioma PC4, conhecido como axioma da coerência, é uma versão mais geral para o axioma PC4', obtido a partir de PC4 quando o evento $C = \Omega$. Note que, deferente de PC4', PC4 é bem definido mesmo quando $P(B) = 0$. Apesar desse caso ser incomum, ele pode ocorrer, como é ilustrado no Exemplo 2.

PC4' Dado 3 eventos, A, B, C, tais que B é não-nulo e $P(B) \neq 0$, então vale $P(A|B) = P(A \cap B)P(B)$.

Exemplo 2. *Suponha que uma moeda é lançada por uma pessoa qualquer. O espaço de possibilidades Ω para a face observada dessa moeda quando ela atingir o chão é $\{cara, coroa\}$. Outro espaço de possibilidades para esse fenômeno é $\Omega' = \{cara, coroa, moeda\ entrar\ em\ órbita\}$. Como essa moeda será lançada por uma pessoa qualquer, a probabilidade do evento $B = \{moeda\ entrar\ em\ órbita\}$ é 0, porém PC4 garante que, por exemplo, as afirmações $P(cara|B) \leq 1/2$ e $P(coroa|B) = 2/3$ sejam coerentes com $P(B) = 0$.*

Apesar do axioma PC4 oferecer essa para vantagem de expressar probabilidade condicional, nesse trabalho será assumido PC1 à PC3 e PC4'. Esse conjunto de axiomas, também conhecido como axiomas de Kolmogorov, será assumido pelo fato de fornecer uma axiomatização na qual a probabilidade da união infinita de eventos estar bem definida.

1.6 Esperança

Dada uma medida de probabilidade $P(\cdot)$ e uma variável aleatória X, ambas definidas para Ω, a *esperança* de X ou *valor esperado* de X é uma combinação entre P e X. Formalmente, o valor esperado de X, denotado por $E[X]$, é calculado por 1.1.

$$E[X] = \sum_{\omega \in \Omega} X(\omega)P(\omega) \tag{1.1}$$

Esse somatório pode ser interpretada como a média, para todo estado da natureza, do valor da variável aleatória X ponderada pelas probabilidades $P(X)$. Nessa média, os valores associados à estados da natureza com mais probabilidade recebem um peso maior. Quando necessário, a notação $E_P[X]$ será usada para explicitar que a medida de probabilidade P foi usada para calcular a esperança de X.

[2] Note que um evento A de probabilidade 0 é diferente do evento vazio. Enquanto o primeiro possui probabilidade bem definida, o segundo, por definição não possui probabilidade alguma.

1.7 Risco e incerteza knightiana

Ao modelar um fenômeno através de um conjunto conjunto credal $\mathcal{K}$, dois casos extremos são possíveis: (i) apenas uma medida de probabilidade é definida, ou seja, $|\mathcal{K}| = 1$; e (ii) $\mathcal{K}$ contém todas as medidas de probabilidade possíveis para Ω, i.e., não é fornecida nenhuma afirmação sobre a ocorrência qualquer evento. Usualmente, o primeiro caso está associado à expressão risco, enquanto o segundo caso está associado à expressão incerteza knightiana, devido aos trabalhos de Knight [Knight, 1921].

Para ilustrar melhor a diferença entre risco e incerteza knightiana, considere a existência de um mecanismo, chamado de *natureza*. Esse mecanismo resolve a *incerteza* associada ao estado da natureza resultante da ocorrência do fenômeno modelado. Por exemplo, durante o lançamento de uma moeda justa, não é possível prever a face que será obtida, porém em algum momento antes da moeda aterrissar, natureza irá escolher qual face será observada como resultado desse lançamento.

Usando essa definição de natureza, quando um fenômeno é modelado através de risco, i.e. $P(\omega)$ é precisamente definido para todo $\omega \in \Omega$, a medida de probabilidade P representa a preferência da natureza na escolha de qual será o seu estado resultante. Nesse ponto fica claro a abordagem subjetivista adotada nesse texto: como não se conhece a preferência exata da natureza, a medida de probabilidade P é a crença de quem modelou o fenômeno em questão sobre essa preferência.

Quando o resultado de um fenômeno é modelado através de incerteza knightiana é considerado ignorância total sobre a preferência da natureza sobre qual será o seu estado após a ocorrência desse fenômeno. Isso porque o conjunto credal $\mathcal{K}$ usado para modelar esse fenômeno irá conter apenas a afirmação $\forall\omega \in \Omega\colon 0 \leq P(\omega) \leq 1$, ou seja, toda medida de probabilidade sobre Ω pertencerá à $\mathcal{K}$. O Exemplo 3 evidência essa diferença entre risco e incerteza knightiana.

Exemplo 3. *Considere que a face observada de uma moeda ao lançá-la é o fenômeno a ser modelado. Para isso, $\Omega = \{cara, coroa\}$ e se não for fornecida nenhuma informação adicional, o conjunto credal $\mathcal{K}$ obtido não terá nenhuma afirmação além das geradas pelos axiomas de probabilidade incondicional. Esse é um cenário de incerteza knightiana, pois todas as preferências possíveis da natureza entre cara e coroa são consideradas possíveis.*

Se alguém acrescentar a afirmação de que para essa moeda vale $P(cara) = 2P(coroa)$, então é obtido um cenário de risco. Isso porque $\mathcal{K}$' conterá apenas a medida de probabilidade $P(cara) = 2/3$ e $P(coroa) = 1/3$. Dessa forma, a pessoa responsável por adicionar essa afirmação está demonstrando acreditar que a natureza tem uma preferência de $2/3$ para cara e $1/3$ para coroa.

A diferença entre esses dois extremos é objeto constante de pesquisas na área de economia e psicologia. Para explicitar o vasto uso desses conceitos em economia, é suficiente citar o discurso de Alan Greenspan de 3 de janeiro de 2004:[3]

> ...incerteza não é só uma característica fortemente presente no cenário político-monetário; ela é uma das características que definem tal cenário. O termo "incerteza" usado aqui abrange ambas "incerteza knightiana" na qual a distribuição de probabilidade dos resultados é desconhecida, e "risco", no qual a incerteza dos resultados é delimitada por uma distribuição de probabilidade conhecida... ([Greenspan, 2004], p. 36)[4]

Até esse momento o termo *incerteza* foi mencionado informalmente, e agora será apresentada uma discussão sobre esse termo e outros termos associados, como *não-determinismo* e *probabilístico*. Gramaticalmente, *não-determinismo* representa a negação do termo determinismo, i.e., a combinação de risco e incerteza knightiana em todas as proporções possíveis. Um fenômeno desse tipo é modelado através de um conjunto credal $\mathcal{K}$, contendo desde uma única medida de probabilidade (risco) até todas as medidas de probabilidade sobre Ω (incerteza knightiana), incluindo todo o espectro entre esses extremos. Nesse texto, como no discurso citado anteriormente de Alan Greenspan, esse caso será chamado de incerteza.

[3] Entre os notáveis trabalhos feitos por Alan Greenspan, está seu cargo como chefe do Banco Central Americano (FED) de junho de 1987 à outubro de 2005.

[4] Tradução livre.

	cinéfilo		atleta	
	ensolarado	chuvoso	ensolarado	chuvoso
parque	5	-10	10	0
cinema	8	10	-5	8
casa	0	0	-10	3

Tabela 1.1: Definição da função de utilidade para os agentes cinéfilo e atleta do Exemplo 4.

Como definição de *não-determinismo*, será considerada a mesma definição usada na teoria da computação para máquinas de Turing não-deterministicas e autômatos não-deterministicos.[5] Informalmente, essa definição está relacionada ao fato da função de transição $\mathcal{T}$ ser multi-valorada, ou seja, dado um estado s e uma entrada σ, o estado resultante dessa máquina é um dos valores devolvidos por $\mathcal{T}$ para $\langle s, \sigma \rangle$. Como não é fornecida nenhuma informação adicional que ajude a prever qual será o estado resultante para $\langle s, \sigma \rangle$, o cenário caracterizado é de incerteza knightiana.

Dessa forma, nesse texto e nas áreas relacionadas à computação, em especial teoria da computação e planejamento em IA, o termo não-determinismo é usado como sinônimo de incerteza knightiana. O mesmo padrão será seguido para o termo *probabilístico*, que é sinônimo de risco para essa áreas.

1.8 Teoria da decisão

De forma simplificada, um *problema de tomada de decisão* pode ser visto como a escolha de uma ação, entre um conjunto de ações possíveis, levando em conta a *utilidade* de cada uma delas. Formalmente, um problema de tomada de decisão (PTD) é uma tupla $\langle \mathcal{S}, \mathcal{A}, U \rangle$ [Giron e Rios, 1980], onde:

PTD1 $\mathcal{S}$ é um conjunto, chamado de espaço de estados, com todos os possíveis estados resultantes da decisão do agente.

PTD2 $\mathcal{A}$ é o conjunto de todas as ações possíveis, chamado de espaço de ações.

PTD3 $U : \mathcal{S} \times \mathcal{A} \in \mathbb{R}$, chamada de função utilidade. Essa função associa uma utilidade a cada ação $a \in \mathcal{A}$ e estado resultante s.

No modelo de um problema de decisão (PTD1 – PTD3), a função utilidade é uma função de ganho que traduz cada par estado e ação em um número real. Dessa forma, dado $s, s' \in \mathcal{S}, a, a' \in \mathcal{A}$, se $U(s, a) > U(s', a')$ então o agente decisor (*decision maker*) prefere $\langle s, a \rangle$ à $\langle s', a' \rangle$. A teoria da utilidade [Cozman, 1997; 2005a; Russel e Norvig, 2003] fornece uma axiomatização que garante a existência U, porém essa teoria tambe é subjetivista, ou seja, diferentes agentes decisores podem possuir diferentes funções de utilidade para um mesmo PTD. O Exemplo 4 ilustra esse fato.

Exemplo 4. *Considere dois agentes decisores, cinéfilo e atleta, e o seguinte problema de tomada de decisão: escolher entre ir para o parque, ir para o cinema ou ficar em casa, sabendo que o dia pode estar ensolarado ou chuvoso. Nesse PTD,* $\mathcal{S} = \{$*ensolarado, chuvoso*$\}$ *e* $\mathcal{A} = \{$*parque, cinema, casa*$\}$ *para ambos os agentes, porém, como é intuitivo, U (Tabela 1.1) é diferente para os agentes cinéfilo e atleta.*

Para cada ação $a \in \mathcal{A}$, é possível definir a função $X_a(s) = U(s, a)$ que representa a utilidade de a para cada estado possível em $\mathcal{S}$. Note que X_a pode ser vista como uma variável aleatória ao assumir $\mathcal{S}$ como espaço de possibilidades $\mathcal{S}$.

Dado um PTD $\langle \mathcal{S}, \mathcal{A}, U \rangle$, se o agente decisor puder especificar uma única medida de probabilidade P sobre $\mathcal{S}$, então ele é conhecido como bayesiano. Nesse cenário de risco, o agente bayesiano avalia cada ação a através de seu valor esperado, ou seja, $E_P[X_a]$ ou simplificando a notação $E_P[a]$. Assim a ação d escolhida é definida por (1.2).

[5] Para uma definição formal de máquinas de Turing não-deterministicas, veja [Lewis e Papadimitriou, 1997]

$$d = \underset{a \in \mathcal{A}}{\operatorname{argmax}} E_P[a] = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \sum_{s \in \mathcal{S}} P(s)U(s,a) \tag{1.2}$$

No entanto, em cenários de incerteza knightiana não é possível aplicar (1.2). Nesse caso, uma suposição comum é que o agente decisor irá fazer usa escolha baseado no pior caso possível. Esse critério é chamado de minimax [Luce e Raiffa, 1957] e é formalmente definido em (1.3) para $\mathcal{K}$ contendo todas as medidas de probabilidade sobre $\mathcal{S}$.

$$d = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \{ \min_{P \in \mathcal{K}} E_P[a] \} = \underset{a \in \mathcal{A}}{\operatorname{argmax}} \{ \min_{P \in \mathcal{K}} \sum_{s \in \mathcal{S}} P(s)U(s,a) \} \tag{1.3}$$

Para o caso mais geral, ou seja, cenários de incerteza, é considerado que o agente decisor possui um conjunto credal $\mathcal{K}$ definido sobre $\mathcal{S}$. O critério de decisão minimax (1.3) também pode ser aplicado para esse caso, e se $\mathcal{K}$ é unitário então (1.3) se reduz à (1.2). Note que, o uso do critério minimax nesse cenário equivale à supor que, após o agente escolher uma ação a, a natureza escolherá como preferência a pior medida de probabilidade $P \in \mathcal{K}$ com relação à $E_P[a]$.

Existem muitas razões para as quais um PTD sob incerteza pode surgir: (i) as crenças do agente são incompletas ou vagas [Levi, 1980; Walley, 1991; 1996], tanto porque não há tempo ou recursos suficientes para elicitá-las, (ii) um grupo de especialistas podem discordar sobre os valores de probabilidades e nenhum acordo, além da coleção da opinião de cada um, pode ser feito [Seidenfeld *et al.*, 1989; Seidenfeld e Schervish, 1990], e (iii) quando o interesse é a robustez das inferências, i.e., na avaliação de quanto variam as inferências quando é permitido que o valor das probabilidades variem [Berger, 1985; Huber, 1980; Kadane, 1984].

Capítulo 2

Uma breve revisão sobre planejamento em inteligência artificial

Estar preparado é metade da vitória.

— Miguel de Cervantes Saavedra, 1547–1616
(escritor espanhol)[1]

Nesse capítulo será apresentada uma breve descrição sobre a área de planejamento e suas principais linhas de pesquisas. A visão unificada que será usada ao longo de todo o texto é baseada em [Bonet e Geffner, 2006; Ghallab *et al.*, 2004; Geffner, 2002]. Também serão apresentados alguns algoritmos para planejamento não-determinístico e probabilístico.

2.1 Definição

Planejamento é o processo de escolha e organização de ações através da antecipação (previsão) de seus efeitos. Esse processo de raciocínio tem como objetivo satisfazer, através da execução das ações escolhidas, algumas metas previamente definidas. Planejamento em IA estuda métodos para automatizar, usando algoritmos independentes de domínio, esse processo de raciocínio. Formalmente, um problema de planejamento pode ser descrito através do *modelo geral de planejamento* (MGP1 – MGP8):

MGP1 $\mathcal{S}$ é um conjunto de estados, chamado de espaço de estados,

MGP2 $s_0 \in \mathcal{S}$ é o estado inicial,

MGP3 $S_G \subseteq \mathcal{S}$ é o conjunto (não-vazio) de estados metas, i.e., os estados que devem ser alcançados pelo sistema,

MGP4 $\mathcal{A}$ é o conjunto de ações, e $\mathcal{A}(s)$ representa as ações aplicáveis no estado s,

MGP5 $\mathcal{E}$ é o conjunto de eventos exógenos, i.e., conjunto de ações que são executadas por fontes externas ao planejador e que não podem ser controladas por ele,[2]

MGP6 $T\colon \mathcal{S} \times \mathcal{A} \times \mathcal{E} \to 2^{\mathcal{S}}$ é a função (geral) de transição de estado, que para cada estado $s \in \mathcal{S}$, ação $a \in \mathcal{A}(s)$ e evento $e \in \mathcal{E}$, devolve um subconjunto S_r de $\mathcal{S}$ representando os estados resultantes possíveis,

[1] Miguel de Cervantes Saavedra é autor da famosa obra *Don Quixote de La Mancha*.

[2] O nome evento exógeno, largamente usado pela comunidade de planejamento em IA, não está associado com a definição de eventos usado na teoria de probabilidades.

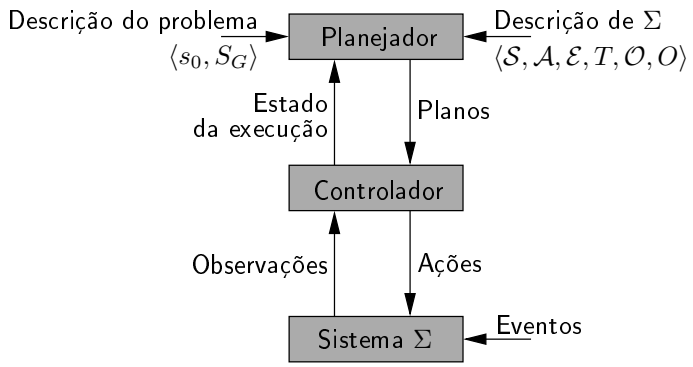

Figura 2.1: Interação entre o planejador, o controlador (entidade que executa os planos) e o sistema que será controlado.

MGP7 $\mathcal{O}$ é o conjunto de observações que o agente pode receber do ambiente, e

MGP8 $O\colon \mathcal{S} \times \mathcal{A} \to 2^{\mathcal{O}}$ é a função de observação que, para cada ação $a \in \mathcal{A}$ e cada estado resultante $s \in \mathcal{S}$, devolve um subconjunto de Ω que representa as observações que o agente pode receber após aplicar a ação a e parar no estado s.

Note que no modelo geral de planejamento, segundo Ghallab *et al.* [2004], os axiomas MGP1 e MGP4 até MGP8 definem um *domínio de planejamento*, i.e., representam um sistema de transição de estados (sistema Σ da Figura 2.1) que simula um ambiente real. Já MGP2 e MGP3 fornecem informações sobre os estados iniciais do ambiente modelado e quais são os objetivos que devem ser atingidos. Em linha gerais, todo *problema de planejamento* consiste em um domínio de planejamento juntamente com informações sobre os estados iniciais e metas à serem alcançadas.

A solução de um problema de planejamento recebe o nome de *plano*. Um plano pode ser simplesmente uma seqüência de ações que devem ser executadas, chamada de plano de malha aberta (*open-loop plan*), ou uma função cuja entrada são as percepções do ambiente num determinado instante e a saída é a ação que deve ser executada naquele instante. Esse último tipo de plano, com retroalimentação, recebe o nome de plano de malha fechada (*closed-loop plan*) e tem como principal exemplo os planos de *contingência*, que são planos com operadores de controle de fluxo, como: `if`, `while` e `until`.

A execução no ambiente (real) das ações contidas em um plano é feita por um controlador. Além disso o controlador também é responsável por intermediar a percepção do ambiente, traduzindo os sinais dos sensores em elementos de $\mathcal{O}$ para serem usados pelo planejador. A interação entre esses três elementos, planejador, controlador e ambiente, está ilustrada na Figura 2.1. Quando não há a necessidade do controlador observar o ambiente nem devolver o estado da execução ao planejador, então o plano computado pelo planejador é do tipo malha aberta. Caso contrário, quando o controlador deve observar o ambiente (mas não necessariamente devolver essa percepção para o planejador), o plano será do tipo malha fechada.

Outro ponto a ser ressaltado é que esse modelo conceitual não estabelece nenhuma linguagem de representação. Logo, a descrição do sistema Σ e os parâmetros do problema (s_0 e S_G) podem ser representados de forma explícita, onde todos os conjuntos e funções descritos pelo modelo geral de planejamento são representados de forma direta na memória. Outra forma de representação é a *implícita*, cuja descrição do modelo é feita através de uma linguagem de alto-nível, como por exemplo PDDL (*Planning Domain Description Language*) [McDermott *et al.*, 1998; Fox e Long, 2003]. Tais linguagens permitem representar estados e ações de forma compacta, através de fluentes, i.e., propriedades do sistema que são alteradas ao longo do tempo, árvores de decisão e outras estruturas de dados. Pesquisas em planejamento requerem uma representação implícita, uma vez que os problemas de interesse envolvem um número muito grande de estados, na ordem de milhões.

2.2 Suposições sobre modelos para planejamento

O modelo geral de planejamento define os componentes que descrevem domínios e problemas de planejamento, porém não é feita nenhuma suposição sobre a estrutura e a relação entre esses componentes. Essas suposições podem ser divididas nos seguintes grupos [Ghallab *et al.*, 2004]:

Cardinalidade de $\mathcal{S}$ (MGP1). Em grande parte dos casos o espaço de estados $\mathcal{S}$ é finito, porém é possível lidar com espaços infinitos. Esse tipo de espaço de estados é necessário quando o problema possui ações que constroem novos objetos ou manipulam variáveis numéricas cujos valores não estão limitados. O uso de espaço de estados infinito geralmente implica na indecidibilidade do problema e perda de qualquer garantia de parada do planejador.

Observabilidade de Σ (MGP7 e MGP8). As observações devolvidas pelo ambiente podem ser suficiente para definir exatamente o estado atual do sistema, ou apenas restringir os estados nos quais ele pode estar. Com base nisso, é possível fazer três suposições sobre os axiomas MGP7 e MGP8, caracterizando um ambiente como:

- Completamente observável, onde não há incerteza sobre o estado atual do mundo, pois o planejador é capaz de determiná-lo através das propriedades observadas do sistema. Essa classe de problemas é caracterizada quando $O(s,a) = O(s',a)$ se e somente se $s = s'$. Uma simplificação pode ser feita usando $\mathcal{O} = \mathcal{S}$ e portanto, $O(s,a) = s$,
- parcialmente observável, no qual as observações feitas pelo planejador permitem apenas criar um conjunto de estados possíveis do ambiente. Esse conjunto, chamado de estado de crença, é construído porque a única inferência que o planejador pode realizar é: se $O(s,a) \neq O(s',a)$ então $s \neq s'$, e
- não-observável, onde o planejador não consegue adquirir informações sobre o ambiente, ou seja, ele não consegue determinar o estado do mundo. Como no ambiente parcialmente observável, o planejador deve considerar um estado de crença, porém, ele deve ser construído apenas com as informações obtidas do estado inicial s_0 e da propagação dos efeitos das ações executadas. Essa classe de problemas é caracterizada quando $O(s,a) = O(s',a) \quad \forall s, s' \in \mathcal{S}$, ou seja, quando Ω é um conjunto unitário.[3]

Dinâmica das ações de Σ (MGP6). O estado resultante após executar uma ação no sistema Σ pode ser modelado através de três dinâmicas diferentes e mutualmente exclusivas, que serão chamadas de dinâmicas básicas [Ghallab *et al.*, 2004]: determinística, não-determinística ou probabilística. A dinâmica determinística estabelece que para cada estado $s \in \mathcal{S}$, evento $e \in \mathcal{E}$ e ação $a \in \mathcal{A}(s)$, existe apenas um estado resultante s' ao aplicar a em s e ocorrer o evento e, ou seja, $T(s,a,e) = S_r = \{s'\}$. Dessa forma, o planejador pode prever o estado resultante da aplicação de uma ação determinística levando em conta os eventos que poderão ocorrer.

Nas dinâmicas não-determinística e probabilística, apesar do sistema Σ sempre estar em um único estado, o planejador não tem informações suficiente para prever exatamente qual é o estado resultante ao aplicar a em s com a ocorrência do evento e. Isso porque o conjunto $S_r \subseteq \mathcal{S}$ devolvido pela função de transição T não é unitário como no caso determinístico. Logo, a única previsão que pode ser feita é que o estado resultante $s' \in S_r$. A diferença entre a dinâmica não-determinística e probabilística é o tipo de incerteza, respectivamente knightiana e risco, isso é, no modelo não-determinístico não é conhecida a medida de probabilidade que rege a escolha de um estado $s' \in S_r$, enquanto no modelo probabilístico é fornecido como parâmetro do problema uma distribuição de probabilidade para cada $T(s,a,e)$ possível, com $s \in \mathcal{S}, a \in \mathcal{A}(s)$e $\in \mathcal{E}$.

Usando as definições acima e os conceitos do Capítulo 1 é possível ver um problema de tomada de decisão seqüencial sob risco como sendo equivalente à um problema de planejamento probabilístico. A mesma relação vale entre tomada de decisão seqüencial sob incerteza knightiana e planejamento não-determinístico.

[3] Quando $\mathcal{O}$ é unitário, $\mathcal{O} = \{\hat{o}\}$, $\hat{o}$ é chamado de observação nula, pois ela não acrescenta nenhuma informação ao planejador.

Dinâmica interna de Σ (MGP5). Um sistema Σ é dito *estático*ambiente!estático quando o conjunto de eventos exógenos $\mathcal{E}$ é vazio, i.e., se o estado resultante de aplicar uma ação permanece inalterado até que outra ação seja executada. Caso $\mathcal{E}$ não seja vazio, então o sistema é dito dinâmico, implicando que entre a execução de duas ações pode ocorrer algum evento exógeno. Esse evento exógeno pode inviabilizar a aplicação da segunda ação.

Caracterização da meta (MGP3). Problemas de planejamento podem ser de *metas restritas*, cujo objetivo é atingir qualquer estado do conjunto S_G (também chamadas de metas de alcançabilidade), ou de *metas estendidas*, onde mais restrições são fornecidas. Entre as metas estendidas mais comuns, estão:

- Otimização de função utilidade, cujo objetivo do planejador é maximizar uma função utilidade U fornecida como um parâmetro do problema. Essa função geralmente é definida através do custo de ações e das recompensas obtidas por atingir determinados estados.
- Otimização de recursos finitos, no qual o planejador deverá se comportar como um escalonador (*scheduler*), pois algumas de suas ações consomem recursos (como por exemplo combustível) que são limitados.
- Restrições de trajetória, onde são adicionadas exigências sobre os estados visitados durante a execução do plano. Essas restrições variam entre estados a serem evitados, estados que obrigatoriamente devem ser visitados, estados onde o sistema deve ser mantido, entre outras.

Note que as diferentes metas estendidas não são mutualmente excludentes, ou seja, mais de uma forma de meta estendida pode surgir na definição de um determinado problema. Além disso, as metas estendidas sofrem interferência direta das outras suposições, como a observabilidade e a dinâmica do sistema. Um exemplo disso é o planejamento para dinâmica de ações probabilística, em que a observação parcial ou completa do ambiente direciona o sistema a encontrar uma solução para o caso médio, i.e. considerando a esperança das variáveis aleatórias envolvidas, enquanto em planejamento não-determinístico o interesse é encontrar a melhor solução no pior caso, ou seja, supondo que a natureza sempre escolherá como seu estado resultante o pior estado possível para o planejador.

Formato da solução. Como foi comentado no início dessa Seção, um plano pode ser de malha aberta, representado por uma seqüência de ações, ou malha fechada. Os planos de malha fechada são representados de forma geral por uma função $\pi\colon \mathcal{O} \to \mathcal{A}$, chamada de política, cuja entrada são as percepções e a saída é a ação que deve ser executada.

Duração das ações. As ações podem ser instantâneas, ou seja, elas não possuem tempo de duração e nesse caso é dito que o sistema possui . Um exemplo de sistema com tempo implícito é o usado *planejamento clássico* (Seção 2.3.1). Em sistemas com *tempo explícito*, o planejador deve levar em conta o tempo de duração das ações. Dessa forma, o planejador deve tratar restrições sobre a interação entre as ações, como por exemplo: (i) ações executadas em paralelo, i.e., quando uma ação pode iniciar ou terminar antes, durante ou depois de outra ação; e (ii) ações que não podem ser executadas simultaneamente. Problemas de planejamento que envolvem ações com duração de tempo se aproximam de problemas de escalonamento, que além de tratarem ações (tarefas) com duração de tempo, também tratam de ações com consumo e produção de recursos.

Interação entre o planejador e Σ. Essa interação pode ser síncrona, i.e., o planejador elabora e executa um plano (geralmente com poucas ações) e, ao receber as observações do ambiente, ele elabora um novo plano, reiniciando o ciclo até atingir a sua meta. Nesse caso o planejador é chamadado de *on-line*, pois o plano solução é construído sob demanda. Quando essa interação ocorre apenas uma vez, ou seja, o planejador calcula uma solução completa *a priori* para o problema e depois a executa, sem necessitar de retroalimentação do controlador, o planejador é chamado de *off-line*. A relação entre a forma da solução e a interação entre o planejador e o ambiente é a seguinte: todo planejador *on-line* gera planos de malha fechada, enquanto todo plano de malha aberta é produzido por planejadores *off-line*.

Quantidade de agentes executores de ações.[4] Um domínio de planejamento é dito de agente único quando os planos enviados ao controlador (Figura 2.1) são executados por apenas um agente. Caso o controlador comande mais de um agente executor, o planejador pode usar essa informação para melhorar alguns parâmetros da execução dos planos, como tempo total gasto, custo total, recursos usados, etc. Nesse caso, chamado de planejamento para multi-agentes, é possível que ações sejam executadas em paralelo, pois diferentes agentes podem executar ações no mesmo instante de tempo. Além disso, quando os agentes executores não são todos iguais, novas restrições podem ser adicionadas ao domínio para modelar a capacidade individual de cada agente.

Com relação as suposições apresentadas anteriormente, nesse trabalho será assumido que todo o domínio de planejamento (sistema Σ) é finito, completamente observável, estático, com tempo implícito e de agente único, enquanto todo problema de planejamento possuirá tanto metas restritas quanto metas baseadas em otimização de função utilidade e solução representável por uma política explícita (plano malha fechada). Além disso, os algoritmos de planejamento propostos nesse trabalho serão tanto *on-line* quanto *off-line*.

A única suposição não definida, é a dinâmica das ações de Σ, que será o objetivo de estudo desse trabalho. Essa suposição será alterada para permitir que um sistema Σ possua, simultaneamente, uma dinâmica tanto não-determinística quanto probabilística que será chamada de dinâmica sob incerteza. Essa alteração, além de tratar as dinâmicas básicas de ações como casos especiais, apresenta uma nova classe de problemas de planejamento que ainda não foi considerada pela comunidade de planejamento em IA.

2.3 Modelos para planejamento

O modelo geral de planejamento (MGP1 – MGP8), não tem como intenção ser um modelo diretamente operacional, mas apenas um referência para a elaboração de modelos mais restritos que possam ser usados em problemas reais. A seguir, será definido o modelo básico de estados para planejamento (MBE1 – MBE6), baseado em [Bonet e Geffner, 2006; 2000], no qual são feitas as suposições discutidas na Seção 2.2. Esse modelo é composto por:

MBE1 $\mathcal{S}$, um conjunto discreto e finito, chamado espaço de estados,

MBE2 $s_0 \in \mathcal{S}$, o estado inicial,

MBE3 $S_G \subseteq \mathcal{S}$, um conjunto não-nulo de estado representando os estados meta,

MBE4 $\mathcal{A}$, o conjunto de ações, e $\mathcal{A}(s)$ representará as ações aplicáveis no estado s,

MBE5 $F(s,a) \subseteq \mathcal{S}$, chamada de função de transição de estados. Essa função mapeia estados s e ações $a \in \mathcal{A}(s)$ em conjuntos não-nulos de estados, i.e. $|F(s,a)| \geq 1$, e

MBE6 $C(s,a) \in \mathbb{R}_+$, chamada de função de custo. Essa função associa o custo de aplicar a ação $a \in \mathcal{A}(s)$ no estado s.

Note que o modelo básico de estados é um caso especial de modelo geral de planejamento, obtido através das seguintes alterações: remover MGP5, MGP7 e MGP8 para satisfazer as suposições de sistema estático e completamente observável; substituir MGP6 por MBE5 para simplificar a representação de transição de estados; e adicionar MBE6 para definir a função utilidade usada como meta estendida. Através do modelo básico de estados é possível definir modelos para as três dinâmicas básicas de ações. Tais modelos são ilustrados na Figura 2.2 e comentados nas próximas subseções.

[4] Essa suposição não faz parte da axiomatização original de [Ghallab *et al.*, 2004], pois ela pode estar implícita nas outras suposições, mas por motivos de clareza ela foi explicitada nesse trabalho.

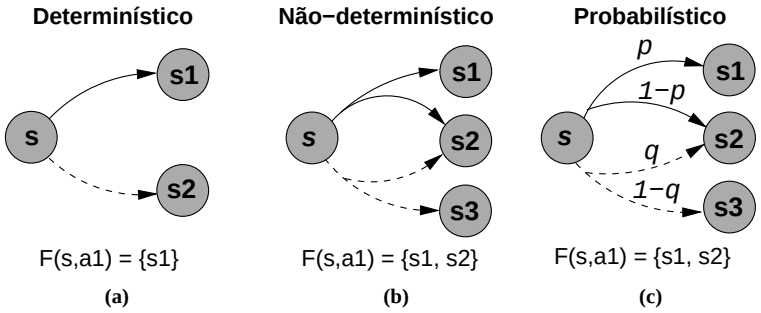

Figura 2.2: Ilustração de diferentes estruturas para a função $F(s,a)$ (MBE5) do modelo básico de estados para planejamento. Arcos contínuos e tracejados representam diferentes ações.

2.3.1 Modelos determinísticos

Os modelos determinísticos, ilustrados na Figura 2.2 (a), são definidos através da seguinte suposição sobre a função de transição (MBE5): $|F(s,a)| = 1$, ou seja, que o efeito de todas as ações é totalmente previsível. Um plano para qualquer modelo determinístico pode ser representado através de uma seqüência de ações $a_0, \ldots, a_{n-1}$. Esse tipo de plano é chamado de válido se e somente se para $0 \leq i \leq n-1, s_{i+1} \in F(s_i, a_i)$, $a_i \in \mathcal{A}(s_i)$ e $s_n \in S_G$.

O modelo determinístico mais difundido em planejamento é o modelo de *planejamento clássico*, no qual, além das suposições assumidas nesse texto, também é feita a restrição adicional de que o custo das ações é uniforme, ou seja, $\forall s \in \mathcal{S}, a \in \mathcal{A}(s)$: $C(s,a) = c > 0$. Nesse modelo, os problemas podem ser vistos como encontrar o menor caminho em grafos dirigidos que comece no estado inicial s_0 e atinja qualquer estado meta $s_G \in S_G$.

Os problemas de planejamento determinístico não são muito próximos da realidade, pois são raros os cenários onde o efeito de cada ação é determinístico. Porém, a solução ótima (de menor custo) pode ser obtida de forma eficiente para alguns jogos e ambientes artificiais, como por exemplo o jogo de *Brigde* [Smith *et al.*, 1996; 1998]. As técnicas de planejamento clássico servem de base para os estudos de planejamento em geral. Através desse modelo, o mais simples da área, foi possível analisar os diferentes espaços de busca [Pereira e Barros, 2004; Penberthy e Weld, 1992], buscas heurísticas [Bonet e Geffner, 2001d; 1999; Hoffmann e Nebel, 2001] e técnicas de controle de buscas [Nau *et al.*, 1999; 2003].

2.3.2 Modelos não-determinísticos

Nos modelos não-determinísticos, representados pela Figura 2.2 (b), a execução de uma ação pode levar o sistema à diferentes estados, sem que o planejador saiba se há alguma preferência entre eles. Esses modelos são a extensão mais simples dos modelos determinísticos, nos quais ações deixam de possuir um único estado resultante possível para possuir um conjunto de estados resultantes possíveis, i.e., $\exists s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$: $|F(s,a)| \geq 1$. Como o planejador não possui uma distribuição de probabilidade representando a preferência da natureza pelos estados resultantes, esses modelos representam problemas sob incerteza knightiana. A solução para problemas dessa classe, usualmente, fornecem alguma garantia sobre o resultado no pior caso possível (estratégia minimax). Entre os possíveis critérios de otimalidade, como por exemplo maximização da probabilidade da meta ser atingida, minimização do caminho até a meta, etc, será escolhido a minimização do custo para atingir a meta. Assim o plano ótima para um problema de planejamento não-determinístico será o plano que possui o menor custo de execução no pior caso (minimax).

Como nesse estudo é assumido observabilidade total, um plano para um problema de planejamento não-determinístico será representado através de uma política π. Essa política pode ser:

- **parcial**, quando π é uma função parcial do estado de estados, i.e., ela não está definida para todo o espaço de estados. Para denotar o domínio de uma política (parcial) π será usado S_π;

- **fechada** em relação a um estado s, se e só se todo estado acessível a partir de s seguindo π estiver contido em S_π, ou seja, se $\bigcup_{s \in S_\pi} F(s, \pi(s)) \subseteq S_\pi$;
- **própria** se e somente se algum estado meta pode ser atingido a partir de todo estado $s \in S_\pi$; e
- **acíclica** em relação a um estado s, se e só se não existe uma seqüência válida de estados t gerada por π, i.e. $t = \langle s_0, \ldots, s_k, \ldots, s_n \in S_G \rangle$ tal que $s_{k+1} \in F(s_k, \pi(s_k))$ para $0 \leq k \leq n-1$, na qual $s_i = s_j$, $0 \leq i < j \leq n$.

Uma política, parcial ou não, para essa classe de problemas é uma solução válida se e somente se ela for própria e fechada com relação a s_0. Toda política válida π associa um valor $V^\pi(s)$ para todo estado $s \in S_\pi$ representando o custo no pior caso de atingir um estado meta partindo do estado s. Essa função V^π, que será comentada em maiores detalhes na Seção 2.3.3, pode ser calculada resolvendo a recorrência (2.1).

$$V^\pi(s) = \begin{cases} 0 & \text{, se } s \in S_G \\ C(s, \pi(s)) + \max\limits_{s' \in F(s,\pi(s))} V^\pi(s') & \text{, caso contrário} \end{cases} \tag{2.1}$$

Para políticas válidas e acíclicas a $V^\pi(s)$ é sempre bem definida, i.e., $\forall s \in S_\pi : V^\pi(s) < \infty$. Seguindo o critério de otimalidade adotado, a política ótima, denotada por π^*, é uma política válida tal que $\nexists \pi, \nexists s \in S_{\pi^*} : V^\pi(s) < V^{\pi^*}(s)$.

2.3.3 Modelos probabilísticos

Os modelos probabilísticos, por exemplo o ilustrados na Figura 2.2 (c), são caracterizados quando as ações possuem efeitos probabilísticos, ou seja, a execução de ações está relacionada ao risco. Em tais modelos, além de existir s e a tal que $|F(s,a)| \geq 1$, também é fornecida, para todo $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$, uma medida de probabilidade condicional $P(\cdot|s,a)$ sobre $F(s,a)$. A solução de problemas representados por esses modelos é representada por uma política que, como um agente bayesiano, deverá maximizar o valor esperado de uma função utilidade, por exemplo o oposto do *custo da política*, a probabilidade de atingir um estado meta, etc.

Processos markovianos de decisão

A principal abordagem para resolver os problemas de planejamento probabilístico é através da sua modelagem como um processo markoviano de decisão (MDP1 – MDP5), conhecido também por MDP (*markovian decision process*) [Howard, 1960; Boutilier *et al.*, 1999]. O modelo dos MDPs foi originalmente proposto pela comunidade de teoria da decisão e fornece um arcabouço (*framework*) capaz de representar problemas de decisão seqüencial sob risco em ambientes completamente observáveis. Devido ao seu poder de expressão e o fato dele ser trivialmente adaptável para ambientes parcialmente e não-observáveis, os MDPs são amplamente usados em IA [Russel e Norvig, 2003].[5]

MDP1 $\mathcal{S}$ é o espaço de estados do sistema, onde será usada a *hipótese de Markov*, ou seja, cada estado possui todas as informações necessárias e suficientes para determinar o efeito de qualquer ação, bem como seu custo;

MDP2 $P_0(\cdot)$ é uma medida de probabilidade sobre $\mathcal{S}$ que define a probabilidade do estado inicial ser $s \in \mathcal{S}$,

MDP3 $\mathcal{A}$ é o conjunto de ações do domínio, e $\mathcal{A}(s)$ representa as ações aplicáveis no estado s,

MDP4 $P(\cdot|s,a)$, para todo $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$, é uma medida de probabilidade condicional sobre $\mathcal{S}$ que define a probabilidade de transição do estado s após aplicar a ação a para cada estado $s' \in \mathcal{S}$;[6]

MDP5 $C : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_+$ é uma função representando o custo de aplicar a ação a no estado s.

[5] A axiomatização dos MDPs pode diferir em relação à tópicos como estado inicial, função utilidade e estados meta. Nesse texto foi adotado uma axiomatização mais próxima da usada em pesquisa operacional.

[6] Uma ação não-aplicável $\hat{a}$ em s recebe probabilidade 0 para todo estado resultante, ou seja, $P(s'|s,\hat{a}) = 0 \quad \forall s' \in \mathcal{S}$.

Para resolver problemas de planejamento probabilístico usando MDPs, primeiro é necessário traduzir MBE2, MBE3 e MBE5 para o modelo dos MDPs. O axioma MBE2 é trivialmente obitido através de MDP2 atribuindo $P_0(s_0) = 1$. Já MBE3 é codificado em MDP4 e MDP5: para todo $s_G \in S_G$ (MBE3) e toda ação $a \in \mathcal{A}(s_G)$, $P(s_G|s_G, a) = 1$ e $C(s_G, a) = 0$. Essa transformação faz com que todos os estados metas (MBE3) tenham o menor custo possível (zero) e sejam estados absorventes, i.e., estados nos quais não se pode sair (beco sem saída). Por último, MBE5 está contido em MDP4: $\forall s \in \mathcal{S}, a \in \mathcal{A}(s), F(s, a) = \{s'|P(s'|s, a) > 0\}$.

Dado um MDP $m = \langle \mathcal{S}, P_0, \mathcal{A}, P, C\rangle$, a seqüência de estados visitados e ações aplicadas durante a execução de uma política é chamada de histórico. A norma de um histórico é mensurada pela quantidade de ações contidas nele, assim $h = \langle s^0, a^0, s^1, a^1, \ldots, a^{n-1}, s^n\rangle$ possui norma n ($|h| = n$). O histórico h é válido se e somente se para todo $0 \leq i < n$: $a^i \in \mathcal{A}(s^i)$ e $P(s^{i+1}|s^i, a^i) > 0$; e o conjunto de históricos válidos ($\mathcal{H}$) recebe o nome de espaço de histórico. Note que o espaço de históricos é um conjunto (possivelmente infinito) enumerável, pois é possível criar uma bijeção entre $\mathcal{H}$ e $\mathbb{N}$.

Uma maneira de classificar MDPs é através do tamanho máximo de seus históricos, denominado horizonte. Um MDP de horizonte finito t ($\forall h \in \mathcal{H}$: $|h| \leq t$) pode ser visto como um problema no qual o agente decisor possui algum recurso finito, como por exemplo energia, bateria ou combustível, que limita a quantidade de ações que ele poderá executar. Porém, existem MDPs nos quais não é possível, ou não se deseja, limitar o tamanho de seus históricos. Esse caso, chamado de horizonte infinito, será assumido como padrão ao longo do desse texto.

O custo de qualquer histórico válido h é representado pela função $V\colon \mathcal{H} \to \mathbb{R}_+$. Essa função é chamada de função valor e, para MDPs de horizonte finito, ela pode ser calculada através do primeiro caso de (2.2). Como a soma simples dos custos pode divergir no caso de MDPs de horizonte infinito, é necessário definir um *fator de desconto* $\gamma \in]0, 1[$ para que a função valor seja bem definida. O fator de desconto pode ser visto como a probabilidade de continuar a execução da política ou, em cenários econômicos, como a inflação decorrente do tempo. A definição de $V(h)$ para horizonte infinito é fornecida pelo segundo caso de (2.2).

$$V(h) = \begin{cases} \sum_{i=0}^{|h|-1} C(s^i, a^i), & \text{para horizonte finito} \\ \sum_{i=0}^{\infty} \gamma^i C(s^i, a^i), & \text{para horizonte infinito} \end{cases} \tag{2.2}$$

Como foi dito no início da seção, a solução usual de modelos probabilísticos, em especial de MDPs, é uma política π. Essa política pode ser *estacionária*, i.e., ela não se altera com o tempo. Nesse caso a melhor ação a ser executada no estado s é sempre a mesma, independente da quantidade de ações que ainda podem ser executadas. Caso contrário, quando a melhor ação pode se alterar com o tempo, a política é chamada de não-estacionária. Uma política não-estacionária é representada pela função $\pi(s, t), s \in \mathcal{S}$ e $t \in \mathbb{N}$, onde s é o estado atual e t é a quantidade de ações que o agente ainda pode executar.

Uma condição suficiente, mas não necessária, para um MDP possuir uma política estacionária como solução ótima é o seu horizonte. Todo o MDP de horizonte infinito possui como solução assintótica uma política estacionária [Boutilier *et al.*, 1999]. Já um MDP de horizonte finito pode ter ou não uma solução ótima representável como uma política estacionária.

Considerando $\mathcal{H}$ como um espaço de possibilidades, é possível calcular a sua probabilidade (incondicional) de um histórico $h \in \mathcal{H}$ através de (2.3). Também é possível calcular essa probabilidade condicionada a uma política π, i.e. $P(h|\pi)$, por (2.4). Note que ambas as medidas dependem apenas de P_0. Dessa forma, a função valor $V\colon \mathcal{H} \to \mathbb{R}$ é uma variável aleatória e sua esperança $E_{P(\cdot|\pi)}[V]$ representa o custo esperado da política π. Para simplificar a notação, $E_{P(\cdot|\pi)}[V]$ será denotado por $E[\pi]$.

$$P(h) = P_0(s^0) \prod_{i=1}^{|h|} P(s^i|s^{i-1}, a^{i-1}) \tag{2.3}$$

$$P(h|\pi) = P_0(s^0) \prod_{i=1}^{|h|} P(s^i|s^{i-1}, \pi(s^{i-1})) \tag{2.4}$$

Note que calcular $E[\pi]$ usando diretamente a definição de valor esperado (2.5), fornecida na Seção 1.6, é muito custoso. Isso porque o espaço de históricos $\mathcal{H}$ é infinito para MDPs de horizonte infinito. Usando a hipótese de Markov, Bellman [1957] elaborou a *função valor para políticas*, uma função recorrente $V^\pi \colon \mathcal{S} \to \mathbb{R}_+$, exibida para MDPs de horizonte infinito em (2.6), que para cada $s \in \mathcal{S}$ representa o custo esperado da política π considerando s como estado inicial do histórico. Dessa forma é possível calcular $E[\pi]$ usando (2.6): $E[\pi] = \sum_{s \in \mathcal{S}} P_0(s) V^\pi(s)$.

$$E[\pi] = \sum_{h \in \mathcal{H}} V(h) P(h|\pi) = P_0(s^0) \sum_{h \in \mathcal{H}} V(h) \prod_{i=1}^{|h|} P(s^i | s^{i-1}, \pi(s^{i-1})) \tag{2.5}$$

$$V^\pi(s) = C(s, \pi(s)) + \gamma \sum_{s' \in \mathcal{S}} P(s' | s, \pi(s)) V^\pi(s') \tag{2.6}$$

Além de poder ser visto como um problema de tomada de decisões seqüenciais, um MDP também pode ser interpretado com um problema de tomada de uma única decisão (PTD). Isso porque, através dos conceitos apresentados nessa seção e na Seção 1.8, é possível descrever o problema de encontrar uma política ótima para um MDP $m = \langle \mathcal{S}, P_0, \mathcal{A}, P, C \rangle$ como o PTD $p = \langle \mathcal{S}, \Pi, -V \rangle$. Nesse PTD p, Π é o *espaço de políticas*, ou seja, o conjunto de todas as políticas de m e para todo $s \in \mathcal{S}$ e $\pi \in \Pi$: $V(s, \pi) = V^\pi(s)$ (2.6). Outro ponto a ser ressaltado é que a função de custo dos MDPs (MDP5) é uma função de perda, ou seja, é oposto da função utilidade, justificando a definição de $-V$ para função utilidade de p.

Como os MDPs modelam um cenário de risco, o planejador usado para resolver m se comportará como um agente bayesiano. Dessa forma, a política ótima (π^*), que é a solução ótima tanto de p quanto de m, é uma política π que maximize $E_{P(\cdot|\pi)}[-V] = -E_{P(\cdot|\pi)}[V] = -E[\pi]$, i.e., que minimize $E[\pi]$. Na Seção 2.4.1 serão apresentados algoritmos para encontrar π^* para MDPs.

Caminho estocástico mínimo

Uma segunda abordagem para resolver problemas de planejamento probabilístico é através da sua modelagem como um problema do caminho estocástico mínimo (SSP1 – SSP6), também chamado de SSP (*stochastic shortest path*) [Bertsekas e Tsitsiklis, 1991]. Um SSP é a extensão direta do problema do caminho mínimo (determinístico) com pesos no qual as ações deixam de ser determinísticas para ser probabilísticas.

SSP1 $\mathcal{S}$ é o espaço de estados do sistema, onde será usada a hipótese de Markov, como em MDP1;

SSP2 $s_0 \in \mathcal{S}$ é o estado inicial do sistema;

SSP3 $S_G \subseteq \mathcal{S}$ é o conjunto de estados meta;

SSP4 $\mathcal{A}$ é o conjunto de ações do domínio, e $\mathcal{A}(s)$ representa as ações aplicáveis no estado s;

SSP5 $P(\cdot|s, a)$ como em MDP4;

SSP6 $C \colon \mathcal{S} \times \mathcal{A} \to \mathbb{R}_+$ representa o custo das ações, como em MDP5.

Os problemas do caminho estocástico mínimo também podem ser vistos como um caso especial dos MDPs. A transformação usada para codificar um SSP com um MDP é a mesma usada na seção anterior para traduzir os axiomas do modelo básico de estados para um MDP. Por isso, os SSPs podem ser vistos como um adaptação dos MDPs para problemas de planejamento. Outro reflexo dessa mudança está no horizonte; enquanto um MDP pode ser de horizonte finito ou infinito, um SSP não necessita de tal conceito pois ele modela um problema de alcançabilidade, i.e., uma vez atingido o estado desejado o problema está resolvido. Como não se deseja impor um limite na quantidade de ações executadas para atingir um estado meta de um SSP, é usada a teoria de MDPs de horizonte infinito para definir SSPs. No entanto, não há a necessidade de usar o fator de desconto γ para os SSPs, pois há uma garantia de que o problema não será executado infinitamente. Com isso, todos os conceitos $V(h), P(h)$ e $P(h|\pi)$ apresentados na seção anterior são definidos da mesma

maneira para um SSP. Já o custo esperado de uma política é definido por $E[\pi] = V^{\pi}(s_0)$ e a função valor para históricos (V^{π}) é definida por (2.7).

$$V^{\pi}(s) = C(s, \pi(s)) + \sum_{s' \in \mathcal{S}} P(s'|s, \pi(s)) V^{\pi}(s') \tag{2.7}$$

A solução de um SSP também é um política, porém ela deve ser apenas fechada em relação à s_0 e própria. Assim, essa política pode ser completa, como nos MDPs, ou parcial, permitindo que ela esteja definida apenas para uma parte do espaço de estados. Essa característica é especialmente desejável em planejamento porque os problemas dessa área são representados de forma implícita, i.e., através de uma linguagem de auto-nível. Por isso, o espaço de estados é reconstruído de forma automática, o que pode resultar em um espaço (geralmente de tamanho exponencial com relação à entrada do problema) onde nem todo o estado é atingível a partir de s_0.

Dado um SSP e o seu MDP equivalente, é possível compara o espaço de estados máximo visitado pelos algoritmos de solução para cada um desses modelos: enquanto o algoritmo para SSPs **no pior caso** visitará todo o *fecho transitivo direto* de s_0, i.e. todos os estados alcançáveis a partir de s_0, o algoritmo para MDPs **necessariamente** deve visitar todo o espaço de estados, pois sua solução é uma politica completa. Como os problemas de planejamento contêm informações sobre o estado inicial e estados metas, é possível reduzir a quantidade de estados visitados ao resolver um MDP. Isso é feito ao substituir o espaço de estados $\mathcal{S}$ do MDP pelo fecho transitivo direto de s_0 em $\mathcal{S}$. Note que os algoritmos de solução de MDPs explorarão **totalmente** esse espaço simplificado, enquanto os algoritmos para SSPs explorarão no **máximo** esse mesmo espaço.

2.4 Algoritmos para planejamento probabilístico

Nessa seção serão exibidos os algoritmos clássicos para encontrar políticas ótimas para MDPs e SSPs. Enquanto os algoritmos para MDPs são desenvolvidos pela comunidade de pesquisa operacional e se baseiam puramente em programação dinâmica, os algoritmos para SSPs são desenvolvidos pela comunidade de IA (em especial de planejamento) e se são baseiam em busca heurística também.

2.4.1 Processos markovianos de decisão

A partir da equação (2.6) é possível formular a subestrutura ótima do problema de encontrar uma política ótima para um MDP, o que possibilita resolvê-lo através de programação dinâmica. Supondo que π^* é uma política ótima, logo a seguinte relação deve ser válida para toda a política π' e todo estado $s \in \mathcal{S}$: $V^{\pi^*}(s) \leq V^{\pi'}(s)$. Assim, a *função valor ótima*, representada por V^*, é simplesmente a função valor associada a qualquer política ótima para esse MDP de horizonte infinito. A substrutura ótima desse problema é chamada de princípio ótimo de Bellman para MDPs ou apenas equação de Bellman [Bellman, 1957] e é exibida em (2.8). Bellman [1957] também provou que a solução da função valor ótima existe e é única.

$$V^*(s) = \min_{a \in \mathcal{A}(s)} \{C(s, a) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s, a) V^*(s')\} \tag{2.8}$$

Dada uma função valor V, a política gulosa π_V associada à V pode ser obtida através de (2.9). Dessa forma, as políticas ótimas de um MDP são as políticas gulosas associada à função valor ótimo, i.e., $\forall s \in \mathcal{S}$: $\pi^*(s) = \pi_{V^*}(s)$. A seguir, serão apresentados dois algoritmos, iteração de valor e iteração de política, desenvolvidos na área de pesquisa operacional.

$$\pi_V(s) = \operatorname*{argmin}_{a \in \mathcal{A}(s)} \{C(s, a) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s, a) V(s')\} \tag{2.9}$$

```
Iteração-de-valor-HI(mdp,γ,ϵ)
  entrada:        mdp, um MDP de horizonte infinito ⟨S, P_0, A, P, C⟩, γ fator de desconto, ϵ,
                  erro máximo permitido entre V e V'.
  saída:          política estacionária π.
  vars. locais:   V, V' funções valor, π política estacionária, a uma ação, δ variação máxima
                  entre as funções de valor.

  V' ← FunçãoValorNula
  repita
     V ← V'
     para cada estado s ∈ S faça
        a ← argmin_{a∈A(s)} {C(s,a) + γ Σ_{s'∈S} P(s'|s,a)V(s')}
        π(s) ← a
        V'(s) ← C(s,a) + γ Σ_{s'∈S} P(s'|s,a)V(s')
  até ||V − V'||_∞ < ϵ(1−γ)/γ
  devolva π
```

Figura 2.3: Algoritmo de iteração de valor para um MDP de horizonte infinito. A cada iteração o algoritmo melhora a sua estimativa da função valor ótima (V^*) até que o erro dessa estimativa seja no máximo ϵ.

Iteração de valor

O princípio ótimo de Bellman é a base do algoritmo de iteração de valor para MDPs de horizonte infinito. O algoritmo, especificado na Figura 2.3, computa a função valor ótima usando programação dinâmica em (2.8). Assim, o procedimento é iniciado com uma função V_0 que atribui um custo esperado inicial, por exemplo 0, para cada estado $s \in \mathcal{S}$ e calcula a função V_{t+1} usando a estimativa V_t da seguinte forma:

$$V_{t+1}(s) = \min_{a \in \mathcal{A}(s)} \{C(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s,a) V_t(s')\}. \tag{2.10}$$

A seqüência de funções $\{V_t\}$ converge linearmente para a função ótima V^* [Puterman, 1994]. Se a freqüência de atualização de cada estado em (2.10) tender ao infinito, então a função V_t convergirá para a função valor ótima (V^*). Do ponto de vista prático, o algoritmo de interação de valor pára quando a norma infinita de $V_t - V_{t+1}$ é suficientemente pequena.[7] Dado um ϵ, representando o erro máximo para a estimativa de V^*, e γ, o fator de desconto do MDP, o critério de parada é ilustrado em (2.11). Esse critério de parada é calculado na prova de convergência do algoritmo de iteração de valor para MDPs de horizonte infinito. Puterman [1994] apresenta essa prova atavés de *reduções* e fornece mais detalhes para a obtenção desse critério de parada. Assim o algoritmo de iteração de valor não possui um limitante superior para a quantidade de iterações que serão executadas, sendo que a complexidade computacional no pior caso de cada uma dessas iterações é $O(|\mathcal{S}|^2|\mathcal{A}|)$ [Papadimitriou, 1994].

$$\|V_t - V_{t+1}\|_\infty = \max_{s \in \mathcal{S}} |V_t(s) - V_{t+1}(s)| < \frac{\epsilon(1-\gamma)}{\gamma}. \tag{2.11}$$

Iteração de política

O algoritmo de iteração de política é baseado no algoritmo de iteração de valor. Nesse algoritmo, ilustrado na Figura 2.4, ao invés de iterativamente melhorar a estimativa do da função valor ótimo, cada iteração melhora diretamente a estimativa da política ótima. A iteração de política recebe uma política inicial arbitrária π_0 e calcula iterativamente a política π_{t+1} baseada em π_t. Cada iteração pode ser dividida em dois passos:

1. **avaliação da política**: Para cada estado $s \in \mathcal{S}$, calcular $V^{\pi_i}(s)$ baseado na política π_i, e

[7] A norma infinita para a função valor, é definida por $\|V\|_\infty = \max_{s \in \mathcal{S}} |V(s)|$.

```
ITERAÇÃO-DE-POLÍTICA(mdp, γ)
  entrada:      mdp, um MDP de horizonte infinito ⟨S, P_0, A, P, C⟩, γ fator de desconto.
  saída:        política estacionária π.
  vars. locais: V^π função valor, Q uma variável real, π política estacionária inicialmente
                aleatória, alterada uma variável booleana

  repita
     V^π ← AVALIAR-POLÍTICA(π,mdp,γ)
     alterada ← FALSO
     para cada estado s ∈ S faça
        Q ← min_{a∈A(s)} {C(s,a) + γ Σ_{s'∈S} P(s'|s,a)V^π(s')}
        V^π(s) ← C(s,π(s)) + γ Σ_{s'∈S} P(s'|s,π(s))V^π(s')
        se Q < V^π(s) então
           π(s) ← argmin_{a∈A(s)} {C(s,a) + γ Σ_{s'∈S} P(s'|s,a)V^π(s')}
           alterada ← VERDADEIRO
  até alterada = FALSO
  devolva π
```

Figura 2.4: Algoritmo de iteração de política. A cada iteração o algoritmo melhora a política π até que ela se estabilize, ou seja, $\forall s \in \mathcal{S}\colon \pi_t(s) = \pi_{t+1}(s)$. A função auxiliar AVALIAR-POLÍTICA calcula o custo esperado da política π baseado em (2.6).

2. **aperfeiçoamento da política**: Para cada estado $s \in \mathcal{S}$, escolher a ação a que minimize (2.12). Se $Q_{i+1}(s,a) < V^{\pi_i}(s)$, então $\pi_{i+1}(s) = a$, senão $\pi_{i+1}(s) = \pi_i(s)$.[8]

$$Q_{i+1}(s,a) = C(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s,a) V^{\pi_i}(s'). \tag{2.12}$$

O critério de parada para o algoritmo de iteração de política é $\forall s \in \mathcal{S}\colon \pi_{i+1}(s) = \pi_i(s)$, ou seja, o algoritmo é executado até que a política não seja atualizada entre uma iteração e outra. No passo 1 é estimado o custo esperado ao executar a política π_i resolvendo o sistema linear $|\mathcal{S}| \times |\mathcal{S}|$ representado por (2.6), o que pode ser computacionalmente caro. Por isso, a complexidade computacional no pior caso de cada iteração (passos 1 e 2) é $O(|\mathcal{S}|^2|\mathcal{A}| + |\mathcal{S}|^3)$, um pouco mais caro do que uma iteração do algoritmo de iteração de valor ($O(|\mathcal{S}|^2|\mathcal{A}|)$) [Papadimitriou, 1994].

No entanto, o algoritmo de iteração de política converge para a política ótima pelo menos linearmente e sobre algumas condições, ele converge super-linearmente [Puterman, 1994]. Dessa forma, a iteração de política requer mais recursos computacionais por iteração do que a iteração de valor, porém sua convergência tende a ser mais rápida, o que representa uma vantagem em alguns domínios.

2.4.2 Caminho estocástico mínimo

Como nos MDPs, é possível formular a equação de subestrutura ótima para SSPs, exibida em (2.13), que é um caso especial do princípio ótimo de Bellman para MDPs. O critério de otimalidade segue a mudança na avaliação do custo médio de uma política, assim uma política π^* é a política ótima se e somente se para toda a política $\pi'\colon V^{\pi^*}(s_0) \leq V^{\pi'}(s_0)$. Caso exista mais de uma política ótima, $\pi_1^*, \cdots, \pi_n^*$, é dado preferência à política definida para o menor domínio (π_i^* tal que para $0 \leq j \leq n$ vale $|S_{\pi_i^*}| \leq |S_{\pi_j^*}|$), pois no caso dos SSPs as políticas podem ser parciais. Note que a norma do domínio de uma política não faz parte do critério de otimalidade.

[8] A função Q, definida em (2.12), recebe esse nome devido seu uso em *Q-learning* [Watkins e Dayan, 1992].

$$V^*(s) = \begin{cases} 0, & \text{se } s \in S_G \\ \min\limits_{a \in \mathcal{A}(s)} \{C(s,a) + \sum\limits_{s' \in \mathcal{S}} P(s'|s,a) V^*(s')\}, & \text{caso contrário} \end{cases} \quad (2.13)$$

Em geral, todo algoritmo para SSPs assumi a hipótese de que pelo menos um estado meta pode ser atingido a partir de qualquer estado s do espaço de estados. Em outras palavras, que todo o nenhum beco sem saída é alcançável a partir de s_0 (exceto se ele for um estado meta). Essa hipótese é conhecida como hipótese de alcançabilidade e sua importância é que, a partir dele é possível provar que existi uma política tal que a probabilidade da execução dessa política terminar em um estado meta é 1, ou seja, ela sempre atingirá um estado meta.

A seguir serão apresentados dois algoritmos que assumem a hipótese de alcançabilidade e são baseados em experimentos (*trial based algorithms*), ou seja, algoritmos nos quais o espaço de estados será explorado através de execuções (ou simulações) ao invés de seqüencialmente.

Programação dinâmica em tempo real

Real Time Dynamic Programming (RTDP) [Barto *et al.*, 1995] pode ser visto como uma versão baseada em experimentos do algoritmo de iteração de valor. Além da avaliação da função valor por experimentos, o RTDP possui outra vantagem sobre o algoritmo de iteração de valor: ele usa uma heurística H para guiar a busca no espaço de estados. Dessa forma, a exploração do espaço de estados feita pelo RTDP pode ser caracterizada como uma exploração gulosa usando a política π_H (a política gulosa definida pela heurística H), o que faz caminhos mais prováveis serem avaliados com mais freqüência do que os menos prováveis.

O algoritmo RTDP, ilustrado na Figura 2.5, assume a hipótese de alcançabilidade para provar que ele não entrará em um ciclo infinito (*loop*) e eventualmente atingirá um estado meta [Barto *et al.*, 1995]. A cada iteração do algoritmo, a função auxiliar `RTDP-Trial` (Figura 2.6) é chamada para computar a ação gulosa em relação a heurística H no estado atual s. No final de cada uma dessas chamadas do `RTDP-Trial`, o valor da heurística $H(s)$, bem como ação $\pi(s)$, são atualizadas. Caso a heurística fornecida inicialmente seja admissível (ou seja, um limitante inferior para V^*), então ela convergirá para a função valor ótima, logo política π também convergirá para a política ótima.

Uma característica interessante do RTDP é o fato dele se comportar com um algoritmo *anytime*, ou seja, ele rapidamente produz uma boa política e gradualmente com o tempo ele a melhora, até que ela convirja para a política ótima. Essa característica torna o RTDP especialmente interessante para problemas nos quais o tempo disponível para resolver o problema é restrito, como por exemplo problemas em robótica e interfaces de interação com pessoas.

Um efeito indesejável da estratégia de exploração do espaço de estados do RTDP é que caminhos improváveis tendem a ser ignorados. Por isso sua velocidade de convergência, que está diretamente ligada à qualidade da heurística H e ao tamanho do espaço de estados, é em geral lenta. Como o algoritmo de iteração de valor, não existe um limitante superior para a quantidade de iterações necessárias para que a função valor ótima seja encontrada. Por isso, é adotado um critério de parada sobre a diferença máxima entre o valor H e o valor da próxima escolha gulosa. Como a solução de um SSP pode ser uma política parcial, o domínio dessa maximização é o conjunto de todos os estados alcançáveis a partir de s_0 seguindo a política atual π, chamado de espaço gerado por s_0 e π ($S_\pi^{s_0}$) e definido formalmente em (2.14).

$$S_\pi^{s_0} = \{s \in \mathcal{S} | s = s_0 \text{ ou } \exists s' \in S_\pi^{s_0} : P(s|s', \pi(s')) > 0\} \quad (2.14)$$

Programação dinâmica em tempo real com rótulos

Motivados pela convergência lenta do RTDP, bem como pela sua falta de um limitante superior para a quantidade de iterações executadas, Bonet e Geffner [2003] adicionaram um procedimento de rotulação de estados ao RTDP, obtendo o algoritmo Labeled Real Time Dynamic Programming (LRTDP) que é ilustrado na Figura 2.7. A idéia do algoritmo é adicionar ruído nas simulações executadas pelo RTDP para favorecer

```
RTDP(ssp, H, ε)
  entrada:      ssp, um SSP ⟨S, s_0, S_G, A, P, C⟩, H, uma heurística admissível para V* e ε,
                erro máximo entre as atualizações de H.
  saída:        política estacionária, própria em relação à s_0, fechada e possivelmente parcial
                π.
  vars. locais: π_H, política gulosa em relação à H.

  repita
      H ← RTDP-Trial(ssp, s_0, H)
      π_H ← Política-Gulosa-Mínima(ssp, s_0, H)
  até  max_{s ∈ S^{s_0}_{π_H}} |H(s) − min_{a∈A(s)} {C(s,a) + Σ_{s'∈S} P(s'|s,a)H(s')}| < ε
  devolva π_H
```

Figura 2.5: Algoritmo RTDP. O algoritmo simula a execução da política gulosa em relação à heurística H (π_H). O algoritmo pára quando a diferença máxima entre o valor de H e o valor da próxima escolha gulosa é menor que ϵ. O método auxiliar RTDP-Trial é ilustrado na Figura 2.6.

```
RTDP-Trial(ssp, s, H)
  entrada:      ssp, um SSP ⟨S, s_0, S_G, A, P, C⟩, s o estado atual e H, uma heurística ad-
                missível para V*.
  saída:        H, heurística admissível para V* atualizada.
  vars. locais: a' a melhor ação para ser executada em s.

  enquanto s ∉ S_G faça
      a ← argmin_{a'∈A(s)} {C(s,a') + Σ_{s'∈S} P(s'|s,a')H(s')}
      H(s) ← C(s,a) + Σ_{s'∈S} P(s'|s,a)H(s')
      s ← Simular(ssp, s, a)
  devolva H
```

Figura 2.6: Método auxiliar do RTDP. Acada iteração, RTDP-Trial se baseia na heurística H para escolher a melhor ação a que deverá ser executada no estado s. Uma vez escolhida a ação, $H(s)$ é atualizado e o estado resultante de aplicar a em s é escolhido aleatoriamente através do método Simular.

caminhos menos prováveis. Assim a velocidade convergência do algoritmo é acelerada, ao mesmo tempo que o comportamento *anytime* é preservado.

Ao invés de desviar o RTDP dos caminhos mais prováveis, o ruído adicionado à simulação evita que os estados nos quais a função valor já convergiu sejam visitados (e atualizados). Esse efeito é obtido através do mecanismo de rotulação que marca um estado s como Resolvido sempre que a atualização de $H(s)$ é considerada satisfatória (menor que um ϵ). Isso é feito através do método VerificarEstadosResolvidos (Figura 2.9), que recebe um estado s e expande, seguindo a política gulosa definida por H, todos os estados que já convergiram e são alcançáveis a partir de s. Assim, VerificarEstadosResolvidos pára sob duas hipóteses: (i) a franja da árvore de busca só contém estados que não convergiram (devolvendo Falso); ou (ii) todos os estados analisados já convergiram (devolvendo Verdadeiro e marcando s como Resolvido). Por isso, se a heurística H é admissível e *monotônica*, i.e. respeita (2.15), então uma chamada do método VerificarEstadosResolvidos, ou rotula um estado como resolvido, ou aumenta o valor de $H(s)$ em pelo menos ϵ, enquanto não diminui o valor de nenhum estado [Bonet e Geffner, 2003].

$$H(s) \leq \min_{a \in \mathcal{A}(s)} \{C(s,a) + \sum_{s' \in \mathcal{S}} P(s'|s,a)H(s')\} \tag{2.15}$$

Como no RTDP, o LRTDP necessita que a hipótese de alcançabilidade seja válida e que a heurística H seja admissível para que o seu método auxiliar LRTDP-Trial (Figura 2.8) não entre em um ciclo infinito. Já a inovação introduzida pelo o LRTDP, um limitante superior para a quantidade de iterações que devem ser executadas até que a heurística H convirja, é dado por Bonet e Geffner [2003] através do seguinte teorema:

```
LRTDP(ssp,H,ε)
  entrada:      ssp, um SSP ⟨S, s_0, S_G, A, P, C⟩, H, uma heurística admissível e monotônica
                para V* e ε, erro máximo permitido para estimar V*.
  saída:        política estacionária, própria em relação à s_0, fechada e possivelmente parcial
                π.
  vars. locais:

  repita
     H ← LRTDP-Trial(ssp,s_0,H,ε)
  até Resolvido(s_0) = Verdadeiro
  devolva Política-Gulosa-Mínima(ssp,s_0,H)
```

Figura 2.7: Algoritmo LRTDP. O algoritmo executa o método auxiliar LRTDP-Trial (Figura 2.8) até que o estado inicial seja marcado como Resolvido. Pela definição do algoritmo, essa condição só é verdadeira quando todos os estados em $S^{s_0}_{\pi_H}$ já convergiram.

```
LRTDP-Trial(ssp,s,H,ε)
  entrada:      ssp, um SSP ⟨S, s_0, S_G, A, P, C⟩, s o estado atual, H, uma heurística admis-
                sível para V* e ε, erro máximo permitido para estimar V*.
  saída:        H, heurística admissível para V* atualizada.
  vars. locais: visitados, pilha de estados visitados sendo o topo o mais recente e a', a melhor
                ação para ser executada em s.

  visitados ← PilhaVazia()
  enquanto Resolvido(s) = Falso faça
     Empilha(visitados, s_0)
     se s ∈ S_G então pare
     a ← argmin_{a'∈A(s)} {C(s,a') + Σ_{s'∈S} P(s'|s,a')H(s')}
     H(s) ← C(s,a) + Σ_{s'∈S} P(s'|s,a)H(s')
     s ← Simular(ssp,s,a)
  enquanto NãoVazio(visitados) faça
     s ← Desempilha(visitados)
     se VerificarEstadosResolvidos(s,H,ε) = Falso então pare
  devolva H
```

Figura 2.8: Método auxiliar do LRTDP responsável por fazer a exploração gulosa do espaço de estados. Como os estados mais próximos do estado meta tendem a convergirem primeiro, o método VerificarEstadosResolvidos (Figura 2.9) é chamado para cada estado visitado, do mais recente ($s_G \in S_G$) até o estado inicial (s_0), até que um estado que ainda não convergiu seja encontrado.

se a hipótese de alcançabilidade for válida e a heurística H for admissível e monotônica, então o LRTDP resolve um SSP em no máximo $\epsilon^{-1} \sum_{s \in \mathcal{S}} \{V^*(s) - H(s)\}$ iterações.

```
VERIFICARESTADOSRESOLVIDOS(ssp, H, s, ε)
  entrada: ssp, um SSP ⟨S, s_0, S_G, A, P, C⟩, H, uma heurística admissível para V*, s, o
           estado atual e ε, erro máximo entre as atualizações de H.
  saída: VERDADEIRO se o estado s convergiu ou FALSO caso contrário.
  vars. locais: convergiu, um variável booleana para marcar se s já convergiu, abertos e
           fechados, pilhas para armazenar (respectivamente) os estados a serem analisados
           e os já analisados, e a, a ação gulosa em relação à H.

  convergiu ← VERDADEIRO
  abertos ← PILHAVAZIA()
  fechados ← PILHAVAZIA()
  se RESOLVIDO(s) = FALSO então EMPILHA(abertos, s)
  enquanto VAZIO(abertos) = FALSO faça
     s ← DESEMPILHA(abertos)
     EMPILHA(fechados, s)
     se H(s) − min_{a∈A(s)} {C(s,a) + Σ_{s'∈S} P(s'|s,a)H(s')} > ε então
        convergiu ← FALSO
        continue
     a ← argmin_{a∈A(s)} {C(s,a) + Σ_{s'∈S} P(s'|s,a)H(s')}
     para cada s' tal que P(s'|s,a) > 0 faça
        se RESOLVIDO(s') = FALSO e CONTEM(abertos ∪ fechados, s') = FALSO então
           EMPILHE(abertos, s')

  se convergiu = VERDADEIRO então
     para cada s' tal que CONTEM(fechados, s')
        MARCARCOMORESOLVIDO(s')
  senão
     enquanto VAZIA(fechados) = FALSO faça
        s ← DESEMPILHA(fechados)
        H(s) ← min_{a∈A(s)} {C(s,a) + Σ_{s'∈S} P(s'|s,a)H(s')}
  devolva convergiu
```

Figura 2.9: Método do LRTDP responsável por rotular os estados que já convergiram e atualizar o valor de H para os que ainda não. Note que devido a possível presença de ciclos nos SSPs, a busca por estados que ainda não convergiram não pode ser feita de forma trivial com um procedimento recursivo das folhas para cima (*bottom-up*).

Capítulo 3

Um modelo para planejamento sob incerteza

É melhor prever mesmo sem certeza
do que não prever nada.

— Henri Poincaré, 1854–1912
(matemático e físico francês)

Nesse capítulo, será apresentado um novo modelo matemático para planejamento capaz de tratar a nova dinâmica de ações proposta nesse trabalho bem como as dinâmicas básicas. Também será provado que esse modelo pode ser traduzido para um *processo markoviano de decisão com probabilidades imprecisas* (*MDPIP – Markov Decision Process with Imprecise Probabilities*) e apresentado alguns algoritmos de solução para o modelo proposto.

3.1 Exemplo de problema de planejamento sob incerteza

Para exemplificar a necessidade de um modelo de planejamento capaz de tratar ações que sejam ao mesmo tempo não-determinísticas e probabilísticas, considere o seguinte exemplo:

Exemplo 5 (Fictício)**.** *Um hospital deseja tratar pacientes cardiopatas, i.e., com doenças cardíacas, através de três procedimentos: administrar uma droga $d1$, administrar outra droga $d2$ ou realizar um transplante cardíaco. Ao administrar algum desses procedimentos, o quadro clínico do paciente pode evoluir para os seguintes estados: cardiopatia controlada (sem efeitos colaterais), cardiopatia controlada com efeitos colaterais, cardiopatia grave, cardiopatia irreversível, AVC (Acidente Vascular Cerebral), ou morte. Os estados cardiopatia controlada, cardiopatia controlada com efeito colateral, AVC e morto são estados terminais, ou seja, definem o escopo do tratamento. Assim, um paciente com um AVC deve ser tratado segundo outro modelo pois o AVC é considerado mais grave do que a cardiopatia; um paciente com cardiopatia controlada (com ou sem efeito colateral) são considerados curados e não necessitam mais de tratamento; enquanto nada mais pode ser feito por um paciente morto. Para simplificar esse exemplo, será considerado que todos os estados podem ser exatamente distinguidos através uma bateria de exames, ou seja, o ambiente será considerado completamente observável.*

Além disso, não há informações exatas sobre a eficiência das drogas $d1$ e $d2$, porque tais medicamentos estão em fase de teste e não se conhece todas as variáveis (referentes aos pacientes) que influenciam o comportamento desses medicamentos. Já o transplante cardíaco, ao contrário desses dois tratamentos, é uma técnica bem difundida e por isso já há uma boa estimativa da sua eficiência. Os dados conhecidos sobre esses três procedimentos são listados na Tabela 3.1.

Estado	Ação	Custo	Prob.	Estados Possíveis
cardiopatia	droga $d1$	30	0,6	cardiopatia controlada cardiopatia controlada com efeito colateral
			0,4	cardiopatia grave cardiopatia irreversível
	droga $d2$	20	0,8	cardiopatia controlada cardiopatia grave
			0,2	cardiopatia irreversível AVC
	transplante	75	0,7	cardiopatia controlada com efeito colateral
			0,3	morto
cardiopatia grave	droga $d1$	30	0,8	cardiopatia grave cardiopatia irreversível
			0,2	AVC
	droga $d2$	25	0,7	cardiopatia com efeito colateral cardiopatia irreversível
			0,3	AVC
	transplante	70	0,6	cardiopatia controlada com efeito colateral
			0,4	morto
cardiopatia irreversível	droga $d1$	40	0,7	cardiopatia irreversível AVC
			0,3	morto
	droga $d2$	30	0,5	AVC
			0,5	morto
	transplante	70	0,6	cardiopatia controlada com efeito colateral
			0,4	morto

Tabela 3.1: Informações fornecidas sobre o efeito e custo de cada um dos tratamentos possíveis para o Exemplo 5. Quando um valor de probabilidade p se refere a mais de um estado, a semântica é que a probabilidade de qualquer um desses estados ocorrem é p.

Para medir o custo de um tratamento, foi adotado como critério a perda de qualidade de vida do paciente após receber esse tratamento. A qualidade de vida de um paciente é uma medida entre zero (morto) à cem (saudável), assim um paciente com cardiopatia irreversível perde 70 pontos de qualidade de vida. Além disso é dado um custo para cada estado terminal, representando quantos pontos de qualidade de vida o paciente perdeu por atingir esse estado terminal. Esse custo é fornecido na Tabela 3.2.

O objetivo do hospital é encontrar uma política π para o tratamento de pacientes cardiopatas que controle a cardiopatia dos pacientes, com ou sem efeitos colaterais, e ainda minimize os pontos de qualidade de vida perdidos devido o tratamento.

O Exemplo 5 define um problema de tomada de decisão seqüencial, pois, talvez seja necessário a aplicação de mais do que um tratamento para atingir um estado terminal. Um caso como esse é ilustrado pelo seguinte histórico: ⟨cardiopatia, droga 2, cardiopatia grave, droga 1, cardiopatia irreversível, transplante, cardiopatia controlada⟩.

Além disso, esse exemplo segue as suposições assumidas na Seção 2.2, ou seja, é um problema com metas simples e baseadas em otimização de função utilidade, com solução representável por uma política explícita e seu domínio é finito, completamente observável, estático, de agente único e possui tempo implícito. Sobre a dinâmica de ações, ela claramente não é determinística, pois um mesmo tratamento aplicado em um determinado estado pode ter diferentes resultados. A seguir são apresentadas duas soluções possíveis para esse exemplo, considerando as duas dinâmicas de ações restantes: não-determinística e probabilística.

Estado terminal	Custo
cardiopatia controlada	0
cardiopatia controlada com efeito colateral	2
AVC	85
morto	100

Tabela 3.2: Custo para o paciente terminar o tratamento em cada um dos estados terminais possíveis do Exemplo 5. Note que ao terminar o tratamento morto, o paciente perde 100 pontos, ou seja, todos os pontos possíveis.

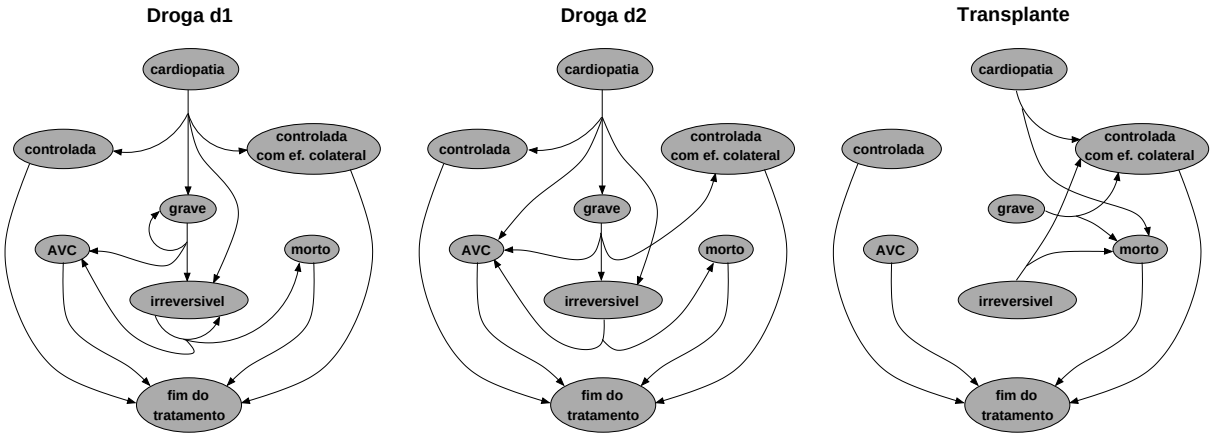

Figura 3.1: Modelo não-determinístico para o Exemplo 5. Se um estado não é origem de arcos, então o tratamento em questão não é aplicável nesse estado.

Solução não-determinística

Nessa solução, as probabilidades contidas na Tabela 3.1 são desconsideradas pois elas não podem ser expressadas através de um modelo não-determinístico. Assim o modelo para o Exemplo 5 é resumido apenas à Figura 3.1. Isso implica em considerar o risco obtido em alguns casos, como por exemplo ao usar a droga $d2$ em cardiopatias irreversíveis ou ao realizar um transplante cardíaco, como incerteza knightiana. Além disso, as outras preferências da natureza (nesse caso o organismo dos pacientes) também serão ignorados e no seu lugar será assumido apenas incerteza knightiana.

Para sinalizar o final do tratamento de forma unificada, foi adicionado o estado `fim do tratamento`, que passa a ser o único estado meta do problema. Assim, o custo dos estados terminais é representado como o custo da ação (determinística) que leva cada estado terminal até o estado fim do tratamento. Esse modelo adaptado foi resolvido usando a Bellman para problemas não-determinísticos (2.1) e a solução obtida ($\pi_{\text{não-det}}$) é apresentada em (3.1).

$$\pi_{\text{não-det}} = \begin{array}{c|c|c} \text{cardiopatia} & \text{cardiopatia grave} & \text{cardiopatia irreversível} \\ \hline \text{droga } d2 & \text{droga } d2 & \text{droga } d2 \end{array} \tag{3.1}$$

Devido o critério do pior caso, o transplante cardíaco sempre representa a pior opção, pois é considerado que o paciente sempre irá falecer e esse é o tratamento com o maior custo. Um efeito similar corre na comparação entre as drogas $d1$ e $d2$. No estado cardiopatia grave, tanto a droga $d1$ como a droga $d2$ irão gerar um AVC no paciente (devido o critério do pior caso), logo a droga $d2$ é escolhida porque seu custo é menor do que o custo da droga $d1$. O mesmo acontece no estado cardiopatia irreversível, onde será considerado que todos os tratamentos irão matar o paciente e novamente a droga $d2$ possui o menor custo. Já no estado cardiopatia, a droga $d2$ é preferida à $d1$ porque, enquanto ela irá gerar um AVC no paciente, a droga $d1$ levará o paciente ao estado cardiopatia grave, estado no qual os melhores tratamentos também irão provocar um

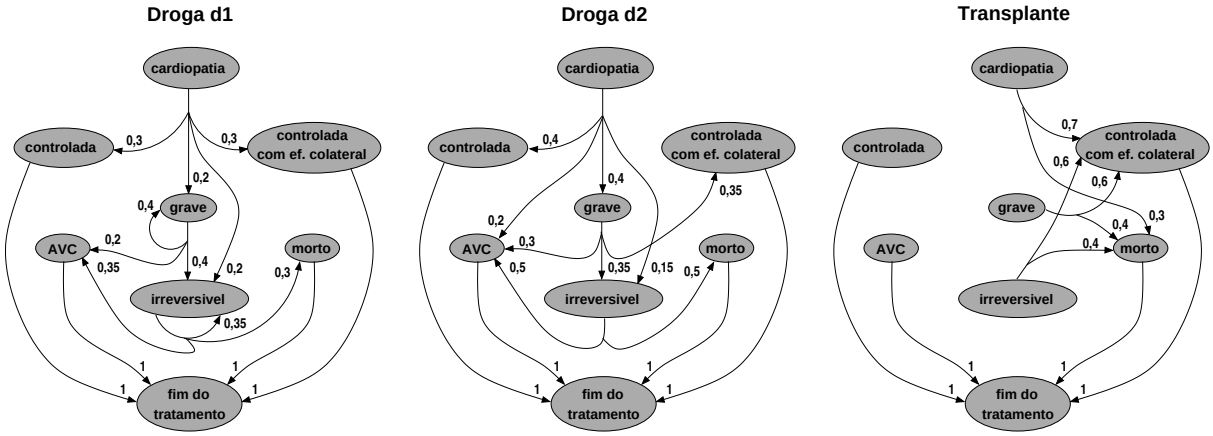

Figura 3.2: Modelo probabilístico para o Exemplo 5. O valor que acompanha cada arco é a probabilidade da transição do estado de origem ao estado de destino ocorrer.

AVC. Assim, aplicar a droga $d2$ no estado cardiopatia terá custo 20 enquanto a droga $d1$ terá custo superior à 30 e ambas, seguindo o critério do pior caso, necessariamente provocarão um AVC no paciente.

Solução probabilística

Nessa solução serão considerados todos os dados fornecidos e também será necessário fazer hipóteses sobre as probabilidades não conhecidas, ou seja, adicionar informações para transformar incerteza knightiana em risco. A maneira mais comum de atribuir valores à essas probabilidades desconhecidas, e que será usada nessa solução, é assumir que seus eventos geradores são equiprováveis. O modelo obtido através dessa hipótese é ilustrado na Figura 3.2.

Como na solução não-determinística, o estado fim de tratamento foi adicionado para simplificar a especificação do estado meta. Dessa forma, a modelagem do exemplo 5 apresentada na Figura 3.2 é um SSP que respeita a hipótese de alcançabilidade. Logo, a política ótima para essa modelagem (π_{prob}) pode ser obtida usando algoritmos como o RTDP e o LRTDP, ou simplesmente resolvendo a equação de Bellman para SSPs (2.13). O resultado obtido é apresentado em (3.2).

$$\pi_{\text{prob}} = \frac{\text{cardiopata} \mid \text{cardiopatia grave} \mid \text{cardiopatia irreversível}}{\text{droga } d1 \mid \text{droga } d2 \mid \text{transplante}} \tag{3.2}$$

Devido o critério do caso médio, inicialmente o transplante cardíaco não é um bom tratamento, mesmo oferecendo grande chances de recuperação do paciente, pois ele é muito custoso. Por isso, apenas no estado cardiopatia irreversível o transplante cardíaco foi escolhido, afinal ele é o único tratamento que pode salvar o paciente. No estado cardiopatia, a droga $d1$ foi escolhida, mesmo sendo mais cara do que a droga $d2$, porque ela apresenta ao mesmo tempo a maior probabilidade de recuperar o paciente (com probabilidade 0,6 o estado resultante será cardiopatia controlada com ou sem efeito colateral) e o melhor pior caso (cardiopatia irreversível). Já no estado cardiopatia grave, a droga $d2$ foi escolhida porque ela apresenta quase a menor probabilidade de terminar o tratamento sem recuperar o paciente (0,3 contra 0,2 da droga $d1$) e é também o tratamento com o menor custo.

Essas diferenças são resultado da técnica usada para trabalhar com as probabilidades parciais fornecidas pelo problema: elas foram ignoradas na solução não-determinística e foram igualmente distribuídas entre seus efeitos na solução probabilística. Ambos os resultados podem ser considerados insatisfatórios. Isso porque a abordagem não-determinística considera que o resultado dos três tratamentos possíveis no estado cardiopatia irreversível será o falecimento do paciente, ignorando que a ação transplante cardíaco tem probabilidade 0,6 de salvar o paciente. Logo, a solução não-determinística apenas tenta minimizar o custo de *matar* o paciente,

condenando-o à morte ao recomendar a droga $d2$ como tratamento. Já a solução probabilística, pode estar assumindo mais riscos do que o necessário. Isso porque o tratamento estabelecido no estado cardiopatia grave é usar a droga $d2$, pois é considerado que a probabilidade de recuperar o paciente é 0,35, mas isso não é necessariamente verdade. Caso o valor real dessa probabilidade seja menor do 0,16, então essa política estará assumindo riscos adicionais, pois nesse caso a própria solução para o caso médio seria escolher o transplante cardíaco. Em alguns cenários, como por exemplo um no qual a vida de um paciente está em jogo, nenhum desses dois efeitos colaterais são admissíveis.

3.2 Um modelo para problemas de planejamento sob incerteza

Para resolver problemas de planejamento sob incerteza será definido o *processo markoviano de decisão com transição valorada por conjuntos* (MDPST — *markovian decision process with set-valued transition*).[1] Os MDPSTs são mais gerais do que os modelos apresentados anteriormente, pois eles fornecem uma semântica precisa para problemas de planejamento com ações não-determinísticas e probabilísticas bem como para a combinação delas em qualquer proporção. Para que isso seja possível, é necessário que o modelo seja capaz de expressar probabilidades parcialmente especificadas, como no Exemplo 5: ao usar a droga $d2$ no estado cardiopatia grave, com probabilidade 0,7 o quadro do paciente evolui para cardiopatia controlada com efeito colateral **ou** cardiopatia irreversível, e com probabilidade 0,3 o paciente sofre um AVC.

Para simplificar a notação, antes de definir essa função mais geral para associar probabilidades será definido *conjunto de estados alcançáveis*. Cada conjunto de estados, unitário ou não, para o qual está associado um valor de probabilidade será chamado de conjunto de estados alcançáveis e denotado por k ($k \subseteq \mathcal{S}$). Assim, no Exemplo 5, o tratamento com a droga $d1$ no estado cardiopatia pode resultar em dois conjuntos de estados alcançáveis distintos: (i) cardiopatia controlada e cardiopatia controlada com efeito colateral; e (ii) cardiopatia grave e cardiopatia irreversível.

Utilizando o conceito de conjuntos de estados alcançáveis, é possível definir uma medida de probabilidade para cada um desses conjuntos, ao invés de cada um dos estados. Em outras palavras, a função de probabilidade condicional $P(s'|s,a)$, antes definida como $P\colon \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$, passa a ser definida por $P\colon 2^{\mathcal{S}} \setminus \emptyset \times \mathcal{A} \times \mathcal{S} \to [0,1]$. Note que ainda há incerteza knightiana sobre o valor de $P(s'|s,a)$ para estado sucessor $s' \in k$. Assim, a semântica dessa nova função de probabilidade é: o conjunto de estados alcançáveis k ($k \subseteq \mathcal{S}$) será o resultado de aplicar a ação a no estado s com probabilidade $P(k|s,a)$, e o estado resultante da natureza s' será escolhido de forma não-determinística entre os estados de k ($s' \in k$).

Há um relação muito estreita entre essa nova função de probabilidade e as *funções de atribuição de massa de probabilidade* associadas com a teoria de capacidades de Choquet de ordem infinita [Shafer, 1976]. Por isso, para distinguir claramente as duas funções de probabilidades, a função de probabilidade para conjuntos de estados alcançáveis será denotada por $m(\cdot|s,a)$ e chamada de função de atribuição de massa (de probabilidade) como na teoria de Choquet de ordem infinita.

Através desses dois novos conceitos, é possível definir um MDPST pelos axiomas MDPST1 – MDPST5. A função de transição de estados para MDPSTs, que será representada por $\mathbf{F}$ para evitar ambiguidades, (MBE5) está implicitamente definida em MDPST4: $\mathbf{F}(s,a) = \{k \subseteq \mathcal{S} | m(k|s,a) > 0\}$. Note que o contra-domínio da função de transição também foi alterada para poder comportar a expressividade dos MDPSTs.

MDPST1 $\mathcal{S}$ é o espaço de estados, onde, da mesma forma como em MDP1, será usada a hipótese de Markov;

MDPST2 $P_0(\cdot)$ é uma medida de probabilidade sobre $\mathcal{S}$ que define a probabilidade do estado inicial $s \in \mathcal{S}$;

MDPST3 $\mathcal{A}$ é o espaço de ações e $\mathcal{A}(s) \subseteq \mathcal{A}$ representa as ações aplicáveis no estado s,

MDPST4 $m(\cdot|s,a)$ uma função de atribuição de massa sobre $2^{\mathcal{S}} \setminus \{\emptyset\}$, na qual $m(k|s,a)$ representa o valor (massa) de probabilidade que deve ser repartido de qualquer maneira entre os elementos de k após aplicar a ação a no estado s.

[1] No primeiro artigo decorrente desse trabalho [Trevizan *et al.*, 2006] o nome do modelo é *planning under uncertainty* (PUU), por questão de clareza, ele foi renomeado para MDPST.

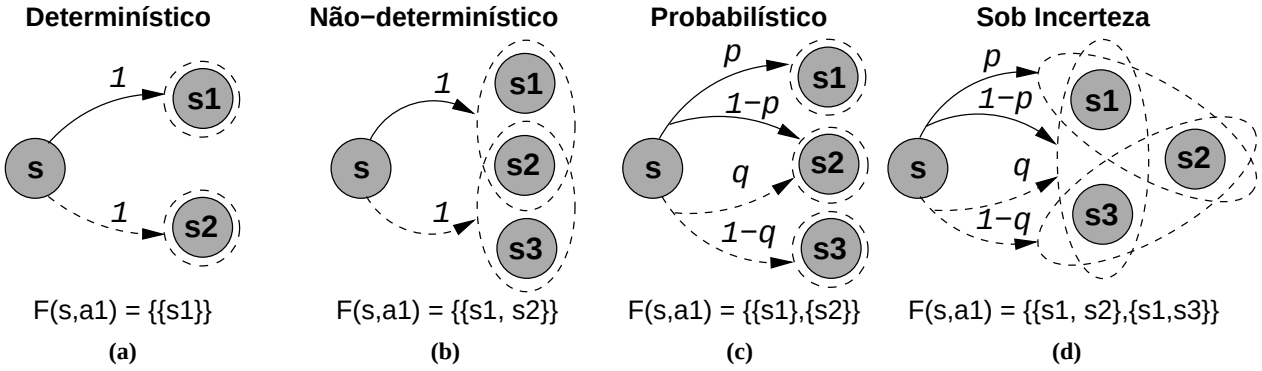

Figura 3.3: Representação gráfica da estrutura da função de transição $\mathbf{F}(s,a)$ implícita em MDPST4 dos MDPSTs. Arcos contínuos e tracejados representam diferentes ações. Círculos contínuos denotam estados e círculos tracejados indicam os conjuntos de estados alcançáveis. Note que a dinâmicas de ações sob incerteza não pode ser representada com o modelo básico de estados (Figura 2.2).

MDPST5 $C : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_+$ é uma função representando o custo de aplicar a ação a no estado s.

Exemplos da representação gráfica dos MDPSTs estão ilustrados na Figura 3.3. É possível comparar o modelo básico de estados (MBE1 – MBE6) e os MDPSTs através das Figuras 2.2 e 3.3. Enquanto o modelo básico de estados não é capaz de expressar os problemas de planejamento sob incerteza (Figura 3.3 (d)), os MDPSTs são capazes de representar todos os modelos contidos no modelo básico de estados (Figuras 3.3 (a), (b) e (c)). Esses modelos são obtidos quando alguma das seguintes restrições não são verdadeiras: (1) $|\mathbf{F}(s,a)| > 1$ e (2) $\exists k \in \mathbf{F}(s,a)$ tal que $|k| > 1$, para $s \in \mathcal{S}, a \in \mathcal{A}(s)$.

Se a primeira restrição é falsa, i.e. $|\mathbf{F}(s,a)| = 1$, e a segunda é verdadeira, então o modelo dos MDPSTs se torna equivalente ao modelo não-determinístico (Figura 3.3 (b)). Isso acontece porque $\forall s \in \mathcal{S}$, $\forall a \in \mathcal{A}(s)$, se $|\mathbf{F}(s,a)| = 1$ então $m(k \in \mathbf{F}(s,a)|s,a) = 1$, o que significa que a escolha de um conjunto de estados alcançáveis será determinística, enquanto a escolha do estado resultante final $s' \in k$ permanecerá não-determinística.

O modelo obtido quando a primeira restrição é verdadeira e a segunda é falsa, i.e., $\forall k \in \mathbf{F}(s,a)$: $|k| = 1$, é equivalente aos MDPs (Figura 3.3 (c)). Isso porque todos os conjuntos de estados alcançáveis pertencentes à $\mathbf{F}(s,a)$ são unitários, então há apenas uma maneira de distribuir a massa de probabilidade $m(k|s,a)$. Assim a escolha não-determinística de um estado resultante final $s' \in k$ será determinística e é possível construir a função $P(\cdot|s,a)$ (MDP4) através da seguinte igualdade: $\forall s \in \mathcal{S}, \forall a \in \mathcal{A}(s), k = \{s'\} \in \mathbf{F}(s,a)$: $P(s'|s,a) = m(k|s,a)$.

Por último, se ambas as restrições forem falsas, o modelo dos MDPSTs será equivalente ao modelo determinístico (Figura 3.3 (a)), pois não haverá mais incerteza na escolha probabilística de um conjunto de estados alcançáveis, nem na escolha não-determinística de um estado resultante final.

A Figura 3.4 ilustra como o problema do Exemplo 5 (Seção 3.1) pode ser descrito através de um MDPST. Note que nenhuma informação foi perdida, como na solução não-determinística, ou acrescentada, como na solução probabilística. A solução para a modelagem como um MDPST será apresentada na Seção 3.5, após serem exibidos alguns algoritmos para resolver MDPSTs (Seção 3.4).

Além das propriedades apresentadas até agora, ainda resta uma propriedade dos MDPSTs a ser discutida: todo MDPST pode ser expresso através de um MDP no qual as probabilidades de transição são imprecisamente conhecidas. Essa conversão será exibida e provada na próxima seção, dando uma semântica formal para os MDPSTs.

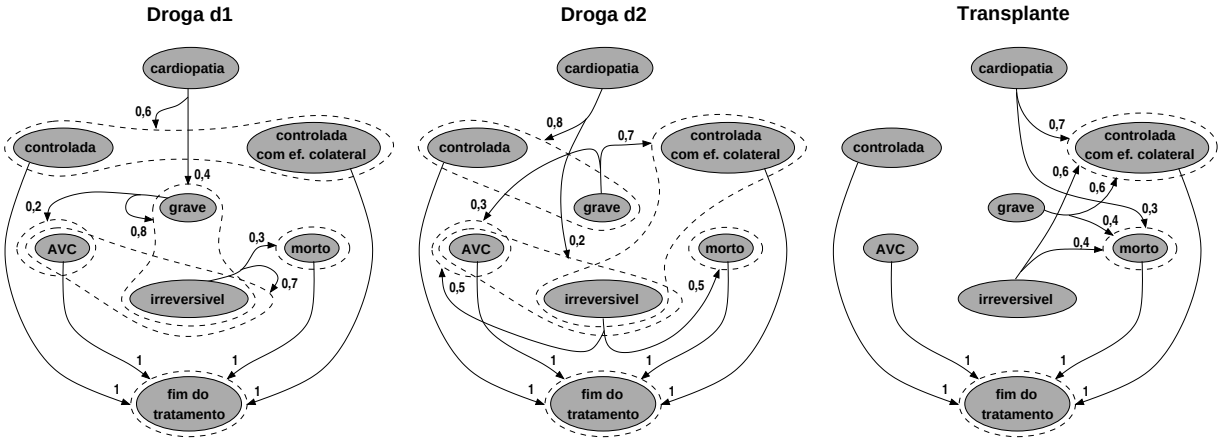

Figura 3.4: Modelagem do Exemplo 5 através de um MDPST. O valor que acompanham cada arco é a massa de probabilidade do conjunto de destino associada a ação e o estado de origem.

3.3 Relação entre MDPST e MDPIP

Um processo de decisão markoviano com probabilidades imprecisas (MDPIP1 – MDPIP5), chamado de MDPIP (*Markov Decision Processes with Imprecise Probabilities*) [White III e Eldeib, 1994; Satia e Lave Jr, 1973] são uma extensão dos MDPs cujas probabilidades que descrevem a transição entre dois estados não são definidas através de números, mas sim por um conjunto de inequações lineares. Consequentemente, os efeitos possíveis de uma ação são modelados através de um conjunto credal $\mathcal{K}$ [Cozman, 2005b] sobre o espaço de estados ao invés de uma distribuição de probabilidade sobre o mesmo espaço. O modelo dos MDPIPs é descrito por:

MDPIP1 $\mathcal{S}$ é o espaço de estados, onde, da mesma forma como em MDP1, será usada a hipótese de Markov;

MDPIP2 $P_0(\cdot)$ é uma função de probabilidade sobre $\mathcal{S}$ que define a probabilidade do estado inicial ser $s \in \mathcal{S}$;

MDPIP3 $\mathcal{A}$ é o espaço de ações e $\mathcal{A}(s) \subseteq \mathcal{A}$ representa as ações aplicáveis no estado s;

MDPIP4 um conjunto credal $\mathcal{K}(\cdot|s,a)$ válido e representável apenas por inequações lineares para expressar todas as possíveis distribuições de probabilidade $P(\cdot|s,a)$ sobre $\mathcal{S}$, e

MDPIP5 $C : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_+$ é uma função representando o custo de aplicar a ação a no estado s.

A interpretação que será adota para essa formulação é baseada em teoria do jogos, e utiliza o conceito de *jogo estocástico* [Shapley, 1953]. Um jogo estocástico é jogo entre duas pessoas no qual a probabilidade de transição de estados é controlada por ambos os jogadores. Na interpretação adotada, um MDPIP é um caso especial de jogo estocástico alternado (assíncrono), cujo o primeiro jogador é o sistema de tomada de decisão (planejador) e o segundo jogador, a natureza, pode assumir qualquer posição entre um adversário ou um advogado.

Para resolver um MDPIP é necessário fazer alguma suposição sobre o comportamento da natureza, e nesse texto será assumido que ela sempre é um adversário para o agente. Assim, uma rodada desse jogo consiste em: (i) o planejador escolhe uma ação a para ser executada no estado s com o objetivo de maximizar o valor de sua função utilidade; e (ii) a natureza escolhe uma distribuição de probabilidade condicional exata em $\mathcal{K}(\cdot|s,a)$ que minimize a função utilidade obtida pelo agente por ter escolhido a.

Como nos MDPs e nos MDPSTs, a função utilidade dos MDPIPs é definida através do custo das ações, ou seja, é uma função de perda. Assim ao minimizar a função utilidade, a natureza está maximizando o custo esperado da política, enquanto o planejador deseja maximizar (minimizar) a função utilidade (o custo esperado). Esse enunciado define um critério minimax, apresentado em (3.3) para horizonte infinito, para encontrar a solução de MDPIPs.

$$f(x) = \begin{cases} 0 & \text{if something} \\ 1 & \text{otherwise} \end{cases} \tag{3.4}$$

$$\mathcal{K}_{\text{cardiop. grave}}(\text{droga } d1) = \left\{ \begin{array}{rcl} 0 \leq & P(\text{cardiop. grave.}|\text{cardiop. grave, droga } d1) & \leq 0.8, \\ 0 \leq & P(\text{cardiop. irrev}|\text{cardiop. grave, droga } d1) & \leq 0.8, \\ 0.8 \leq & \sum_{\substack{s \in \{cardiop.grave., \\ cardiop.irrev.\}}} P(s|\text{cardiop. grave, droga } d1) & \leq 0.8, \\ 0.2 \leq & P(\text{AVC}|\text{cardiop. grave, droga } d1) & \leq 0.2 \end{array} \right\}$$

$$\mathcal{K}_{\text{cardiop. irrev.}}(\text{droga } d1) = \left\{ \begin{array}{rcl} 0 \leq & P(\text{cardiop. irrev.}|\text{cardiop. irrev, droga } d1) & \leq 0.7, \\ 0 \leq & P(\text{AVC}|\text{cardiop. irrev, droga } d1) & \leq 0.7, \\ 0.7 \leq & \sum_{\substack{s \in \{AVC, \\ cardiop.irrev.\}}} P(s|\text{cardiop. irrev, droga } d1) & \leq 0.7, \\ 0.3 \leq & P(\text{Morto}|\text{cardiop. irrev, droga } d1) & \leq 0.3 \end{array} \right\}$$

Figura 3.5: Modelagem através de um MDPIP da ação droga $d1$ do Exemplo 5. A função de probabilidade $P(s'|s,a)$ representa a probabilidade do estado s' ser o estado resultante ao aplicar a ação $a \in \mathcal{A}(s)$ no estado s. Note que essa modelagem contém a mesma quantidade de informações fornecida na Tabela 3.1.

$$V(s) = \min_{a \in \mathcal{A}(s)} \max_{P(\cdot|s,a) \in \mathcal{K}(\cdot|s,a)} \{C(s,a) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s,a)V(s')\} \tag{3.3}$$

Em [Satia e Lave Jr, 1973] é provado que a solução para (3.3), representada por $V^*(s)$ e chamada de princípio ótima de Bellman para MDPIPs, existe e é única. Também é demonstrado que a política ótima para um MDPIP de horizonte infinito pode ser expressa através de um política estacionária (Seção 2.4).

A Figura 3.5 exibe a modelagem dos tratamentos possíveis para o estado cardiopatia grave do Exemplo 5 (Seção 3.1) como ações de um MDPIP. Como na modelagem usando um MDPST, nenhuma informação foi perdida ou acrescentada. A Proposição 1, apresentada a seguir, prova que todo MDPST pode ser convertido para um MDPIP.

Proposição 1. *Todo MDPST pode ser escrito como um MDPIP.*

Demonstração. Note que, a diferença na axiomatização de um MDPST para um MDPIP é apenas a função que representa a incerteza das ações. Assim a prova se reduz a demonstrar que MDPST4 implica em MDPIP4.

Primeiro, note que MDPST4 limita a probabilidade do estado resultante ser s' depois de aplicar a ação a no estado s por (3.5). Isso é devido a definição de conjunto de estados alcançáveis: se para todo cada $k \in \mathbf{F}(s,a)$, $s' \notin k$, então a natureza não pode escolher s' como efeito não-determinístico de a.

$$0 \leq m(\{s'\}|s,a) \leq P(s'|s,a) \leq \sum_{\substack{k \in \mathbf{F}(s,a) \\ s' \in k}} m(k|s,a) \leq \sum_{k \in \mathbf{F}(s,a)} m(k|s,a) = 1 \tag{3.5}$$

Definição 1. *Seja $\mathcal{D}(k,s,a)$, definido apenas para $k \in \mathbf{F}(s,a)$, um subconjunto de $\mathcal{S}$ definido por* (3.6). *Esse conjunto representa todos os efeitos não-determinísticos de $k \in \mathbf{F}(s,a)$ que não pertencem a nenhum outro conjunto de estados alcançáveis $k' \in \mathbf{F}(s,a)$ para $k \neq k'$.*

$$\mathcal{D}(k,s,a) = k \setminus \bigcup_{\substack{k' \in \mathbf{F}(s,a) \\ k' \neq k}} k' \tag{3.6}$$

A partir de MDPST4 é possível obter para todo $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$ dois limitantes para $m(k|s,a)$, onde $k \in \mathbf{F}(s,a)$: um limitante inferior através da soma das probabilidades de cada estado pertencente à k, e

um limitante superior através da soma das probabilidades de cada estado pertencente à $\mathcal{D}(k,s,a)$. Esse dois limitantes são apresentados em (3.7).

$$0 \leq \sum_{s' \in \mathcal{D}(k,s,a)} P(s'|s,a) \leq m(k|s,a) \leq \sum_{s' \in k} P(s'|s,a) \leq 1 \tag{3.7}$$

Dado um determinado $s \in \mathcal{S}$ e um $a \in \mathcal{A}(s)$, o conjunto das inequações (3.5) e (3.7) descrevem $\mathcal{K}(\cdot|s,a)$. Assim, o conjunto das inequações (3.5) e (3.7) para todo o $s \in \mathcal{S}$ e todo $a \in \mathcal{A}(s)$ decrevem o conjunto credal $\mathcal{K}$ definido em MDPIP4.

□

Corolário 2. *Se, para um MDPST m, existe $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$ tal que $\bigcap_{k \in \mathbf{F}(s,a)} k = \emptyset$, então* (3.7) *pode ser simplificado para* (3.8).

$$m(k|s,a) = \sum_{s' \in k} P(s'|s,a) \tag{3.8}$$

Demonstração. Como para $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$ vale $\bigcap_{k \in \mathbf{F}(s,a)} k = \emptyset$, então $\mathcal{D}(k,s,a) = k$, logo os dois extremos de (3.7) são iguais. □

Definição 2. *Dado um MDPST $q = \langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$, o MDPIP $r = \langle \mathcal{S}, P_0, \mathcal{A}, \mathcal{K}, C \rangle$ obtido através da Proposição 1 é chamado de MDPIP associado à q.*

A Proposição 1 torna os resultados de [White III e Eldeib, 1994; Satia e Lave Jr, 1973] válidos para MDPSTs e, em especial, estabelece (3.3) como princípio ótimo de Bellman para o critério minimax adotado. Note que o MDPIP associado a um MDPST não usa toda a expressividade permitida por MDPIP4, pois as incógnitas das inequações geradas ($P(\cdot|s,a)$) são sempre multiplicadas pelo fator 1. Além disso, cada conjunto credal $\mathcal{K}(\cdot|s,a)$ do MDPIP associado conta com apenas duas classes de inequações: (i) para todo $s'\mathcal{S}$, um limitante inferior e superior para a probabilidade de $P(s'|s,a)$ (3.5); e (ii) para todo $k \in \mathbf{F}(s,a)$, um limitante inferior e superior para a somatória de $P(\cdot|s,a)$ sobre, respectivamente, k e $\mathcal{D}(k,s,a) \subseteq k$ (3.7).

Devido a essa estrutura dos MDPIPs associados a MDPSTs, é possível simplificar o princípio ótimo de Bellman (3.3) para o caso dos MDPSTs. Esse resultado, que gera o princípio ótimo de Bellman para MDPSTs, é fornecido no Teorema 3.

Teorema 3. *Para qualquer MDPST $q = \langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$ e seu MDPIP associado $r = \langle \mathcal{S}, P_0, \mathcal{A}, \mathcal{K}, C \rangle$, o princípio ótimo de Bellman para MDPIPs* (3.3) *é equivalente à:*

$$V^*(s) = \min_{a \in \mathcal{A}(s)} \{C(s,a) + \gamma \sum_{k \in \mathbf{F}(s,a)} m(k|s,a) \max_{s' \in k} V^*(s')\} \tag{3.9}$$

Demonstração. Veja o Apêndice A. □

A conseqüência imediata do Teorema 3 é a redução da ordem de complexidade de algoritmos para resolver especificamente MDPSTs. Para exibir uma análise para o pior caso do cálculo de $V^*(s)$, é necessário limitar superiormente para todo $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$ a norma de $\mathbf{F}(s,a)$ de um MDPST $q = \langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$. Esse limitante para q, representado por $\overline{\mathbf{F}}$, é definido por $\overline{\mathbf{F}} = \max_{s \in \mathcal{S}} \{\max_{a \in \mathcal{A}(s)} |\mathbf{F}(s,a)|\}$. No pior caso, $\overline{\mathbf{F}} = 2^{|\mathcal{S}|} - 1$, porém esse é um caso raro que modela um MDPST no qual, para um estado s e uma ação $a \in \mathcal{A}(s)$, é especificado $m(k|s,a) > 0$ para todo subconjunto k de $\mathcal{S}$ (exceto para $k = \{\emptyset\}$).

Dado um MDPST, a maneira mais intuitiva de resolvê-lo é aplicar o princípio ótimo de Bellman para o MDPIP associado à ele. Nessa abordagem, cada iteração feita para calcular V^* resolve, para todo $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$, o sistema linear contido em $\mathcal{K}(\cdot|s,a)$ (Proposição 1) para encontra a medida de probabilidade $P(\cdot|s,a)$ que simula a escolha da natureza. Como esse programa linear possui $|\mathcal{S}|$ variáveis e seu tamanho é proporcional à $\overline{\mathbf{F}}$, a ordem de complexidade de cada iteração é $O(|\mathcal{S}|^{P+Q}|\mathcal{A}|\overline{\mathbf{F}}^Q)$, para $P \geq 2$ e $Q \geq 1$. O valor de P e Q está associado ao algoritmo de programação linear usado. Assim, por exemplo, o uso do

algoritmo de pontos interiores [Kojima *et al.*, 1988] implica em $P = 6$ e $Q = 1$ e o uso do algoritmo de Karmarkar [Karmarkar, 1984] instancia P como 3.5 e Q como 3.

Usando o princípio ótimo de Bellman para MDPSTs (3.9) fornecido pelo Teorema 3, a ordem de complexidade de cada iteração do cálculo de $V^*(s)$ é $O(|\mathcal{S}|^2|\mathcal{A}|\overline{\mathbf{F}})$. Isso porque, a medida de probabilidade que maximiza o lado direito de (3.3) é representada pela escolha $\max_{s' \in k} V(s')$ de (3.9), dispensando a necessidade que calculá-la através de um programa linear.

Dessa forma, a ordem de complexidade para encontrar a função valor ótimo de um MDP $q = \langle \mathcal{S}, P_0, \mathcal{A}, P, C \rangle$ e do seu MDPST $r = \langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$ equivalente é a mesma: $O(|\mathcal{S}|^2|\mathcal{A}|)$ [Papadimitriou, 1994]. Esse resultado é devido $\mathbf{F}(s, a)$ (de r) ser definida da seguinte maneira: $\forall s \in \mathcal{S}, a \in \mathcal{A}(s)$: $\mathbf{F}(s, a) = \{\{s_0\}, \{s_1\}, \ldots, \{s_n\}\}$ tal que $P(s_i|s, a) > 0$ para $0 \leq i \leq n$. Com essa suposição sobre $\mathbf{F}(s, a)$, $\overline{\mathbf{F}}$ será limitada em $|\mathcal{S}|$ para todo MDP, e a escolha $\max_{s' \in k} V(s')$ será feita em tempo constante.

Na próxima seção serão apresentados algoritmos para resolver MDPSTs baseados na Proposição 1 e no Teorema 3. Para efeito de comparação, também será fornecido o algoritmo de iteração de política que não faz uso do Teorema 3.

3.4 Algoritmos de solução para MDPSTs

Nessa seção serão fornecidos algoritmos para encontrar políticas ótimas para MDPSTs usando o critério minimax definido na Seção 3.3. Os algoritmos, iteração de valor e iteração de política, são uma adaptação de suas versões para MDPs (Seção 2.4) feita através da Proposição 1 e do Teorema 3. Também é apresentado uma versão do algoritmo de iteração de política que não usa o Teorema 3. Por último será exibido como adaptar os algoritmos de SSPs para o respectivo caso especial dos MDPSTs.

3.4.1 Iteração de política baseado apenas em MDPIPs

Devido à Proposição 1, todo algoritmo usado para encontrar políticas para MDPIPs também pode ser usado em MDPSTs. No entanto, esse processo não é imediato, pois é necessário adaptar alguns conceitos principais que diferem entre os dois modelos. Para exemplificar a aplicabilidade de algoritmos desenvolvidos para MDPIPs em MDPSTs, foi selecionado uma versão modificada do algoritmo de iteração de política (Seção 2.4) desenvolvido por Satia e Lave Jr [1973]. Esse algoritmo, como o algoritmo de iteração de política para MDPs, é divido em duas fases: (i) avaliação da política, na qual o valor da função valor é calculado, e (ii) aperfeiçoamento da política, que gera uma política cujo valor da função valor é menor do que a original ou o algoritmo é terminado.

O algoritmo de iteração de política modificado é ilustrado na Figura 3.6. Note que esse algoritmo é similar ao algoritmo de iteração de política para MDPs (Figura 2.4), sendo a principal diferença o cálculo e uso dos conjuntos credais $\mathcal{K}(\cdot|s, a)$. A adaptação do algoritmo está na implementação da função auxiliar AVALIAR-POLÍTICA-SATIA (Figura 3.7). No algoritmo original ela se resume a resolver (ou estimar) o princípio ótimo de Bellman para MDPs (2.6), porém nessa versão para MDPSTs é necessário primeiro encontrar uma medida de probabilidade que satisfaça a suposição feita sobre a escolha da natureza. Após encontrar a medida de probabilidade que satisfaz o critério minimax ($P_{\max}$), a função AVALIAR-POLÍTICA-SATIA calcula o custo esperado da política da mesma forma que em um MDP.

3.4.2 Iteração de valor e iteração de política

Tanto o algoritmo de iteração de valor, quanto o algoritmo de iteração de política (Seção 2.4), são obtidos de forma imediata ao substituir o princípio ótimo de Bellman para MDPs (2.6) pela sua versão para MDPSTs (3.9). Os algoritmos obtidos são ilustrados, respectivamente, na Figura 3.8 e Figuras 3.9.

O ganho computacional gerado pelo Teorema 3 é evidenciado ao se comparar as duas versões do algoritmo de iteração de política para MDPSTs fornecidas (Figuras 3.6 e 3.9). Para encontrar a política ótima, o algoritmo ITERAÇÃO-DE-POLÍTICA-SATIA e seu método auxiliar AVALIAR-POLITICA-SATIA resolvem o programa linear induzido pela escolha, para todo $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$, de $P(\cdot|s, a) \in \mathcal{K}(\cdot|s, a)$ em cada iteração. Como no

```
Iteração-de-política-Satia(mdpst, γ)
 entrada:        mdpst, um MDPST de horizonte infinito ⟨S, P_0, A, m, C⟩, γ fator de desconto.
 saída:          política estacionária π.
 vars. locais:   V^π função valor, K um vetor de restrições lineares para armazenar o conjunto
                 credal K, a_antiga uma ação, π política estacionária inicialmente aleatória,
                 alterada uma variável booleana

 para todo s ∈ S faça
    para todo a ∈ A(s) faça
       K(·|s,a) ← Construir-Conjunto-Credal(mdpst,s,a)

 repita
    V^π ← Avaliar-Política-Satia(π, V^π,mdpst,γ,K)
    alterada ← Falso
    para cada estado s ∈ S faça
       a_antiga ← π(s)
       π(s) ← argmin_{a∈A(s)} max_{P(·|s,a)∈K(·|s,a)} {C(s,a) + γ Σ_{s'∈S} P(s'|s,a)V^π(s')}
       se a_antiga ≠ π(s)(s) então
          alterada ← Verdadeiro
 até alterada = Falso
 devolva π
```

Figura 3.6: Algoritmo de iteração de política adaptado para MDPSTs usando o critério minimax adotado. A cada iteração o algoritmo melhora a política π até que ela se estabilize, ou seja, $\forall s \in \mathcal{S}\colon \pi_t(s) = \pi_{t+1}(s)$. A função auxiliar Avaliar-Política-Satia é ilustrada na Figura 3.7.

algoritmo na versão para MDPs (Figura 2.4), o algoritmo de iteração de política para MDPSTs não explicita $P(\cdot|s,a) \in \mathcal{K}(\cdot|s,a)$, pois essa medida de probabilidade está implícita na escolha $\max_{s' \in k} V(s')$.

3.4.3 O problema do caminho mínimo para MDPSTs

Como nos MDPs, é possível definir um modelo que represente o caso especial dos MDPSTs nos quais o estado inicial é único e é fornecido um conjunto de estados metas. Esse problema será chamado de problema do caminho mínimo com transição valorada por conjuntos (SPST1 – SPST6), ou simplesmente SPST (*shortest path with set-valued transition*). A teoria de SSPs pode ser facilmente adaptada para SPSTs usando o Teorema 3, resultando no princípio ótimo de Bellman para SPSTs ilustrado em (3.10).

SPST1 $\mathcal{S}$ é o espaço de estados do sistema, onde será usada a hipótese de Markov, como em MDP1;

SPST2 $s_0 \in \mathcal{S}$ é o estado inicial do sistema;

SPST3 $S_G \subseteq \mathcal{S}$ é o conjunto de estados meta;

SPST4 $\mathcal{A}$ é o conjunto de ações do domínio, e $\mathcal{A}(s)$ representa as ações aplicáveis no estado s;

SPST5 $m(\cdot|s,a)$ como em MDPST4;

SPST6 $C\colon \mathcal{S} \times \mathcal{A} \to \mathbb{R}_+$ representa o custo das ações, como em MDPST5.

$$V^*(s) = \begin{cases} 0, & \text{se } s \in S_G \\ \min\limits_{a \in \mathcal{A}(s)} \{C(s,a) + \sum\limits_{k \in \mathbf{F}(s,a)} m(k|s,a) \max\limits_{s' \in k} V^*(s')\}, & \text{caso contrário} \end{cases} \tag{3.10}$$

O algoritmo `RTDP-MDPST`, ilustrado na Figura 3.10, também é obtido de forma imediata ao substituir (2.8) por (3.10) no algoritmo `RTDP` (Figura 2.5). Note que o mesmo pode ser feito para a versão rotulada do RTDP,

AVALIAR-POLÍTICA-SATIA(π,mdpst,γ,$\mathcal{K}$)
entrada: π, uma política estacionária, mdpst, um MDPST de horizonte infinito $\langle \mathcal{S}, P_0, \mathcal{A}, m, C\rangle$, γ fator de desconto, $\mathcal{K}$, o conjunto credal obtido pela Proposição 1.
saída: Função valor V^π.
vars. locais: P e $P_{\max}$, matrizes $|\mathcal{S}| \times |\mathcal{S}| \times |\mathcal{A}|$ para armazenar a medida de probabilidade de transição de estados, e V^π função valor.

```
para todo s ∈ S faça
    P(·|s,π(s)) ← SORTEAR-MEDIDA-DE-PROBABILIDADE(K(·|s,a))

repita indeterminadamente
    V^π ← ESTIMAR-FUNÇÃO-VALOR(mdpst,P)
    para todo s ∈ S faça
        P_max ← argmax_{P'(·|s,π(s))∈K(·|s,π(s))} {C(s,π(s)) + γ Σ_{s'∈S} P'(s'|s,π(s))V^π(s')}
    se ∀s ∈ S: V^π(s) = C(s,π(s)) + γ Σ_{s'∈S} P_max(s'|s,π(s))V^π(s') então
        interrompa a repetição
    senão
        P ← P_max
devolva V^π
```

Figura 3.7: Algoritmo de avaliação de políticas para MDPSTs sem usar o Teorema 3. A função auxiliar ESTIMAR-FUNÇÃO-VALOR estima, usando aproximações sucessivas, o custo esperado da política π baseado em (2.6) e no valor de P.

o LRTDP, mantendo os mesmos benefícios obtidos para SSPs: maior velocidade de convergência; e limitante superior na quantidade de iterações necessárias para a função valor convergir (Seção 2.4.2).

Porém, provar que um SPST respeita a hipótese de alcançabilidade, garantindo que o RTDP e o LRTDP não entra em um ciclo infinito (*loop*) deixa de ser trivial. Isso porque a hipótese de alcançabilidade, que todo o nenhum beco sem saída é alcançável a partir de s_0 (exceto se ele for um estado meta), deve ser válida para qualquer medida de probabilidade escolhida pela natureza no caso dos SPSTs.

O primeiro reflexo dessa nova restrição é que deve existir pelo menos um estado $\hat{s} \in \mathcal{S}$, uma ação $a \in \mathcal{A}(\hat{s})$ e um conjunto de estados alcançáveis $k_G \subseteq S_G$ tal que $m(k_G|\hat{s}, a) > 0$. Caso contrário, a natureza poderá escolher uma medida de probabilidade tal que nenhum estado meta seja atingido, i.e. $\forall s \in \mathcal{S}, \forall a \in \mathcal{A}(s), \forall s_G \in S_G$: $P \in \mathcal{K}$ tal que $P(s_G|s, a) = 0$, pois em (3.5) o limitante inferior para $P(s_G|s, a)$ é 0 para esse caso. Porém isso não é suficiente para provar a hipótese de alcançabilidade, pois resta provar que esse estado $\hat{s}$ pode ser alcançado. A Proposição 4 fornece uma condição suficiente para provar a hipótese alcançabilidade para SPSTs.

Proposição 4. *Dado um SPST $q = \langle \mathcal{S}, s_0, S_G, \mathcal{A}, m, C\rangle$, se para todo estado $s \in \mathcal{S}$, toda ação $a \in \mathcal{A}(s)$, e todo estado $s' \in \bigcup_{k\in\mathbf{F}(s,a)}$ vale $m(\{s'\}|s, a) > 0$, então basta provar que a hipótese de alcançabilidade vale* ***para pelo menos uma escolha qualquer de probabilidade*** *da natureza.*

Demonstração. Seja $\mathcal{K}$ o conjunto credal do MDPST (Proposição 1) associado ao SPST $q = \langle \mathcal{S}, s_0, S_G, \mathcal{A}, m, C\rangle$ e $P \in \mathcal{K}$ um medida de probabilidade qualquer. Se a hipótese de alcançabilidade for verdadeira para o SSP $r = \langle \mathcal{S}, s_0, S_G, \mathcal{A}, P, C\rangle$, então existe uma política π tal que para todo histórico $h_\pi = \langle s^0 = s_0, \pi(s^0), s^1, \ldots, \pi(s^{n-1}), s^n \in S_G\rangle$ induzido por π vale:

$$\prod_{i=1}^{|h|} P(s^i|s^{i-1}, \pi(s^{i-1})) > 0 \tag{3.11}$$

Como, para $1 \leq i \leq n$, $m(\{s^i\}|s^{i-1}, \pi(s^{i-1})) > 0$ por hipótese, então $P(s^i|s^{i-1}, \pi(s^{i-1})) > 0$ em (3.5). Logo a natureza não consegue escolher um medida de probabilidade tal que (3.11) seja exatamente zero e portanto existe pelo menos uma política (π) que sempre atinge o estado meta.

```
ITERAÇÃO-DE-VALOR-HI-MDPST(mdpst,γ,ϵ)
  entrada:      mdpst, um MDPST de horizonte infinito ⟨S, P_0, A, m, C⟩, γ fator de desconto,
                ϵ, erro máximo permitido entre V e V'.
  saída:        política estacionária π.
  vars. locais: V, V' funções valor, π política estacionária, a uma ação, δ variação máxima
                entre as funções de valor.

  V' ← FunçãoValorNula
  repita
     V ← V'
     para cada estado s ∈ S faça
        a ← argmin_{a∈A(s)} {C(s,a) + γ Σ_{k∈F(s,a)} m(k|s,a) max_{s'∈k} V(s')}
        π(s) ← a
        V'(s) ← C(s,a) + γ Σ_{k∈F(s,a)} m(k|s,a) max_{s'∈k} V(s')
  até ||V − V'||_∞ < ϵ(1−γ)/γ
  devolva π
```

Figura 3.8: Algoritmo de iteração de valor para um MDPST de horizonte infinito usando o critério minimax adotado. A cada iteração o algoritmo melhora a sua estimativa da função valor ótima (V^*).

□

Note que, para um SSP q expressado como um SPST r (r só possui efeitos probabilísticos) a condição de alcançabilidade para r será equivalente à de q e a condição exigida pela Proposição 4 é sempre válida.

3.5 Solução do Exemplo 6 como um MDPST

A modelagem para o Exemplo 5 fornecida na Figura 3.4 define um SPST cujo estado inicial é cardiopatia e o estado meta é fim do tratamento. A solução ótima para esse SPST, π_{st} (3.12), é uma mistura da política ótima para a solução não-determinística (3.1) e para a solução probabilística (3.2). Por exemplo, no estado cardiopatia a droga $d1$ é escolhida, como na solução não-determinística, pois ela é um escolha mais conservadora; já no estado cardiopatia irreversível, o transplante cardíaco é o tratamento escolhido, como na solução probabilística, pois ele é o único tratamento capaz de controlar a cardiopatia do paciente apesar de ter o pior caso igual ao dos outros tratamentos (a morte do paciente). A seguir será apresentado uma comparação entre as três políticas obtidas nesse trabalho: $\pi_{\text{não-det}}$ (3.1), π_{prob} (3.2), π_{st} (3.12).

$$\pi_{\text{st}} = \begin{array}{c|c|c} \text{Cardiopata} & \text{Cardiopatia grave} & \text{Cardiopatia irreversível} \\ \hline \text{droga } d1 & \text{Transplante} & \text{Transplante} \end{array} \tag{3.12}$$

3.5.1 Comparação com as soluções anteriores

Como o Exemplo 5 define um cenário de incerteza, é necessário estender alguns conceitos definidos na Seção 2.3.3 para que seja possível comparar duas ou mais políticas. O primeiro deles é a medida de probabilidade de um histórico $h \in \mathcal{H}$ condicionada a uma política $\pi \in \Pi$ ($P(h|\pi)$). Para um MDP $q = \langle \mathcal{S}, P_0, \mathcal{A}, P, C \rangle$, $P(h|\pi)$ é definido por (2.4), ou seja, essa medida de probabilidade depende apenas de P_0 e $P(\cdot|s,a)$. Porém, nem sempre há uma única medida de probabilidade em um MDPST, assim, $P(\cdot|\pi)$ deve ser definido por (3.13).

$$P(h|\pi) = P_0(s^0) \prod_{i=1}^{|h|} P(s^i|s^{i-1}, \pi(s^{i-1})),\ P \in \mathcal{K} \tag{3.13}$$

ITERAÇÃO-DE-POLÍTICA-MDPST(mdpst, γ)
entrada: mdpst, um MDPST de horizonte infinito $\langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$, γ fator de desconto.
saída: política estacionária π.
vars. locais: V^π função valor, Q uma variável real, π política estacionária inicialmente aleatória, *alterada* uma variável booleana

repita
  $V^\pi \leftarrow$ AVALIAR-POLÍTICA-MDPST(π,mdpst,γ)
  alterada $\leftarrow$ FALSO
  para cada estado $s \in \mathcal{S}$ faça
    $Q \leftarrow \min_{a \in \mathcal{A}(s)} \{C(s,a) + \gamma \sum_{k \in \mathbf{F}(s,a)} m(k|s,a) \max_{s' \in k} V^\pi(s')\}$
    $V^\pi(s) \leftarrow C(s,\pi(s)) + \gamma \sum_{k \in \mathbf{F}(s,\pi(s))} m(k|s,\pi(s)) \max_{s' \in k} V^\pi(s')$
    se $Q < V^\pi(s)$ então
      $\pi(s) \leftarrow \operatorname{argmin}_{a \in \mathcal{A}(s)} \{C(s,a) + \gamma \sum_{k \in \mathbf{F}(s,a)} m(k|s,a) \max_{s' \in k} V^\pi(s')\}$
      alterada $\leftarrow$ VERDADEIRO
até *alterada* = FALSO
devolva π

Figura 3.9: Algoritmo de iteração de política para MDPSTs usando o critério minimax adotado. A cada iteração o algoritmo melhora a política π até que ela se estabilize, ou seja, $\forall s \in \mathcal{S}$: $\pi_t(s) = \pi_{t+1}(s)$. Diferente da função AVALIAR-POLÍTICA-SATIA (Figura 3.7), AVALIAR-POLÍTICA-MDPST usa (3.9) para calcular o custo esperado da política π.

Como essa nova definição de $P(h|\pi)$ depende da escolha da natureza, dois casos especiais são interessantes: o valor mínimo e máximo de $P(h|\pi)$. A probabilidade mínima de um histórico h dado π, representada por $\underline{P}(h|\pi)$, é calculado considerado que a natureza deseja minimizar, para todo i, a probabilidade de transição $P(s^i|s^{i-1}, \pi(s^{i-1}))$. Esse valor, também chamado de probabilidade inferior, é definido formalmente por (3.14). De forma análoga, é possível calcular a probabilidade máxima (ou superior), representada por $\overline{P}(h|\pi)$.

$$\underline{P}(h|\pi) = P_0(s^0) \prod_{i=1}^{|h|} \min_{P(\cdot|s^{i-1},\pi(s^{i-1})) \in \mathcal{K}(\cdot|s^{i-1},\pi(s^{i-1}))} P(s^i|s^{i-1}, \pi(s^{i-1})) \tag{3.14}$$

O conceito de probabilidade inferior e superior sobre $\mathcal{H}$ dado uma política π pode ser propagado para a definição de valor esperado de π (2.5). Assim, o custo esperado mínimo de π é definido como $\underline{E}[\pi] = E_{\underline{P}(\cdot|\pi)}[V] = \sum_{h \in \mathcal{H}} V(h) \underline{P}(h|\pi)$. Note que V nesse caso é a função valor de um histórico ($V : \mathcal{H} \to \mathbb{R}_+$) definida em (2.2) e que independe da medida de probabilidade. Definindo a função valor mínima de uma política 3.15, baseada no princípio ótimo de Bellman para MDPSTs (Teorema 3), é possível calcular $\underline{E}[\pi]$ de forma mais eficiente por (3.16). Os mesmos resultados valem para o custo esperado máximo $\overline{E}[\pi]$, usando $\max_{s' \in k}$ no lugar de $\min_{s' \in k}$ em (3.15) para definir $\overline{V}^\pi(s)$. Note que, devido a conversão de um SPST para MDPST, $\underline{E}[\pi]$ pode ser simplificado no caso dos SPSTs para apenas $\underline{E}[\pi] = \underline{V}^\pi(s_0)$ e $\overline{E}[\pi]$ também pode ser simplificado da mesma forma.

$$\underline{V}^\pi(s) = C(s,\pi(s)) + \gamma \sum_{k \in \mathbf{F}(s,\pi(s))} m(k|s,\pi(s)) \min_{s' \in k} \underline{V}^\pi(s') \tag{3.15}$$

$$\underline{E}[\pi] = \sum_{s \in \mathcal{S}} P_0(s) \underline{V}^\pi(s) \tag{3.16}$$

A Figura 3.12 apresenta o intervalo $[\underline{E}[\pi], \overline{E}[\pi]]$ para todas as políticas possíveis ao modelar o Exemplo 5 como um SPST (Figura 3.4). O gráfico foi gerado computando, para cada política $\pi \in \Pi$, $\underline{E}[\pi] = \underline{V}^\pi(s_0)$ e $\overline{E}[\pi] = \overline{V}^\pi(s_0)]$, usando o algoritmo de iteração de valor para SPSTs e $\epsilon = 10^{-5}$. Note que nesse exemplo,

RTDP-SPST(spst, H, ϵ)
entrada: spst, um SPST $\langle \mathcal{S}, s_0, S_G, \mathcal{A}, m, C \rangle$, H, uma heurística admissível para V^* e ϵ, erro máximo entre as atualizações de H.
saída: política estacionária, própria em relação à s_0, fechada e possivelmente parcial π.
vars. locais: π_H, política gulosa em relação à H.

repita
$H \leftarrow$ RTDP-Trial-SPST(spst, s_0, H)
$\pi_H \leftarrow$ Política-Gulosa-Mínima(spst, s_0, H)
até $\max_{s \in S^{s_0}_{\pi_H}} |H(s) - \min_{a \in \mathcal{A}(s)} \{C(s,a) + \sum_{k \in \mathbf{F}(s,a)} m(k|s,a) \max_{s' \in k} H(s')\}| < \epsilon$
devolva π_H

Figura 3.10: Algoritmo de RTDP para SPSTs. O algoritmo simula a execução da política gulosa em relação à heurística H (π_H) considerando a natureza como um adversário (critério minimax). O algoritmo pára quando a diferença máxima entre o valor de H e o valor da próxima escolha gulosa é menor que ϵ. O método auxiliar RTDP-Trial-SPST é ilustrado na Figura 3.11.

RTDP-Trial-SPST(spst, s, H)
entrada: spst, um SPST $\langle \mathcal{S}, s_0, S_G, \mathcal{A}, m, C \rangle$, s o estado atual e H, uma heurística admissível para V^*.
saída: H, heurística admissível para V^* atualizada.
vars. locais: a' a melhor ação para ser executada em s.

enquanto $s \notin S_G$ faça
$a \leftarrow \operatorname{argmin}_{a' \in \mathcal{A}(s)} \{C(s,a') + \sum_{k \in \mathbf{F}(s,a')} m(k|s,a') \max_{s' \in k} H(s')\}$
$H(s) \leftarrow C(s,a) + \sum_{k \in \mathbf{F}(s,a)} m(k|s,a) \max_{s' \in k} H(s')$
$s \leftarrow$ Simular(spst, s, a)
devolva H

Figura 3.11: Método auxiliar do RTDP para SPSTs. Acada iteração, RTDP-Trial-SPST se baseia na heurística H e na hipótese que a natureza é um adversário (critério minimax) para escolher a melhor ação a que deverá ser executada no estado s. Uma vez escolhida a ação, $H(s)$ é atualizado e o estado resultante de aplicar a em s é escolhido aleatoriamente através do método Simular.

o espaço de políticas Π possui 27 elementos, pois os estados cardiopatia, cardiopatia grave e cardiopatia irreversível possuem 3 ações aplicáveis cada, enquanto os outros estados possuem apenas 1 ação aplicável.

Como era previsto, o intervalo do custo esperado da política π_{st} exibe a característica minimax: é a solução com o melhor pior caso. Em outras palavras, devido o critério minimax adotado nesse trabalho, toda a solução para um MDPST obtida através dos algoritmos fornecidos terá o menor custo máximo.

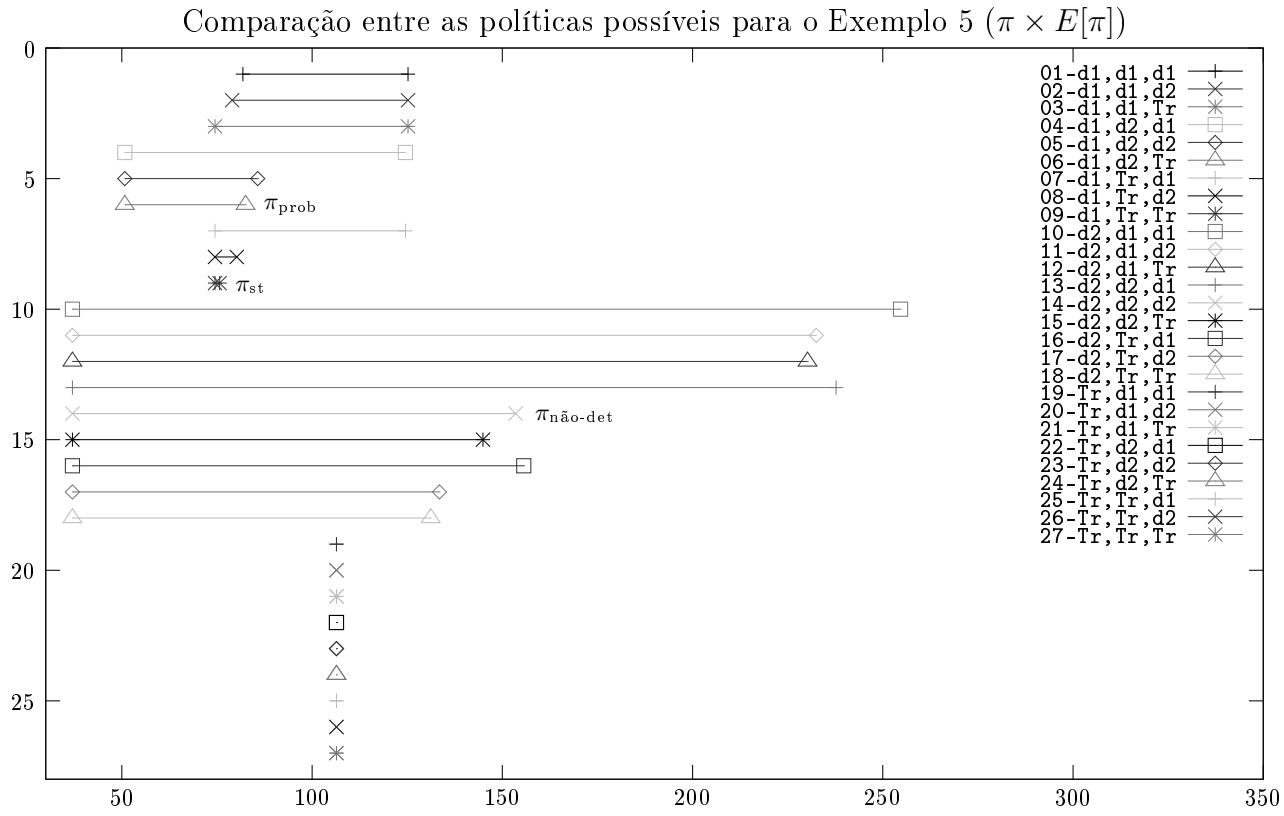

Figura 3.12: Gráfico com o intervalo do custo esperado para todas as políticas do Exemplo 5. Sua legenda exibe as ações que devem ser executadas, respectivamente, nos estados cardiopatia, cardiopatia grave e cardiopatia irreversível. As representação das ações é: d1 para droga $d1$, d2 para droga $d2$ e Tr para transplante cardíaco.

Capítulo 4

Trabalhos correlatos

... o método de Gauss-Seidel era aparentemente desconhecido para Gauss e não recomendado por Seidel, ...

— Berman e Plemmons [1979] sobre o método de Gauss-Seidel para iteração de valor.

Existem muitos trabalhos considerados como referências principais sobre planejamento determinístico [Bonet e Geffner, 2001d; Hoffmann e Nebel, 2001; Kautz e Selman, 1999; Russel e Norvig, 2003], não-determinístico [Bertoli *et al.*, 2001] e probabilístico [Boutilier *et al.*, 1999; Bonet e Geffner, 2000; 2006]. Apesar de alguns trabalhos definirem abordadens unificadas para todas essas diferentes dinâmicas [Bonet e Geffner, 2006; Ghallab *et al.*, 2004], não há nenhum trabalho que apresente um modelo capaz de tratar e resolver problemas de planejamento sob incerteza.

Os principais planejadores desenvolvidos pela comunidade de planejamento, como por exemplo o GPT [Bonet e Geffner, 2001a], são capazes de resolver problemas para apenas as três dinâmicas básicas de ações. Tais sistemas são compostos por um conjunto de algoritmos, por exemplo, o GPT usa o algoritmo HSP [Bonet e Geffner, 2001d; 2001b] para problemas determinísticos com observação total, A* para problemas determinísticos e não-determinísticos com observação nula e LRTDP para os outros casos. Em todos esses sistemas os problemas de planejamento sob incerteza são negligenciados.

Na área de planejamento, o modelo mais próximo ao proposto nesse trabalho é o BMDP (*Bounded-parameter MDP*) apresentado em [Givan *et al.*, 2000]. Esse modelo é fruto dos trabalhos de redução de modelo e agregação de estados feitos por Robert Givan, Thomas Dean e Sonia Leach [Dean *et al.*, 1997], cujo objetivo é gerar de forma automática um modelo menor (com menos estados) que seja equivalente ao modelo original. Como critério de equivalência de MDPs é usado o conceito de bissimulação estocástica [Givan *et al.*, 2003].

Os BMDPs também são um caso especial dos MDPIPs cuja semântica é representar um conjunto de MDPs (de parâmetros exatos). Nos BMDPs, a probabilidade de estar no estado s' após aplicar a ação a no estado s ($P_\beta(s'|s,a)$) é representada por um limitante inferior ($\underline{P}_\beta(s'|s,a)$) e superior ($\overline{P}_\beta(s'|s,a)$). Uma das limitações dos BMDPs com relação aos MDPIPs, também aos MDPSTs, é que os BMDPs não são capazes de expressar inequações referentes a soma da probabilidade de transição para diferentes estados (por exemplo (3.7)).

Outro modelo proposto na comunidade de planejamento é o MDP Algébrico (*Algebraic MDP*) [Perny *et al.*, 2005]. Os MDPs Algébricos, AMDPs, são uma representação genérica de MDPs para qualquer estrutura de recompensa e de incerteza. A única restrição feita nesse modelo é que cada uma dessas estruturas devem ser expressadas através de um semi-anel. Apesar da generalidade do modelo, ele só foi explorado para problemas de horizonte finito, sendo sua aplicabilidade em casos de horizonte infinito um dos tópicos propostos pelos autores para pesquisas futuras.

Na área de pesquisa operacional, em que foi desenvolvido o modelo MDPIP, existe um literatura muito breve sobre MDPIPs [Satia e Lave Jr, 1973; White III e Eldeib, 1994; Harmanec, 1999], porém nenhum desses trabalhos fornecem a semântica de problemas de planejamento sob incerteza explicitada na presente proposta. Em [Satia e Lave Jr, 1973] é apresentado pela primeira vez esse modelo e é fornecido o algoritmo de iteração de política para MDPIPs. Além disso, esse trabalho também fornece uma abordagem Bayesiana para o modelo.

Em [White III e Eldeib, 1994] é feito uma análise sobre os limitantes para a função valor ótima (V^*) e também são apresentados dois novos algoritmos para a solução de MDPIPs: (i) uma versão de revisão de recompensas para MDPIPs e (ii) uma versão do algoritmo de iteração de política modificada [Puterman e Shin, 1978]. O primeiro algoritmo usa operadores algébricos de redução para, iterativamente, obter uma aproximação melhor de V^*. Já o segundo algoritmo, usado a técnica de iterações sucessivas para calcular a função valor entre as iterações (3.3).

O último artigo [Harmanec, 1999] fornece uma visão diferente de uma solução para um MDPIP. Enquanto os trabalhos anteriores, bem como essa proposta, buscam soluções minimax, i.e. considerar que a natureza é um adversário do agente decisor, em [Harmanec, 1999] é usado um critério de preferência entre políticas. Esse critério de dominância de políticas é baseado no custo esperado superior e inferior de cada política. Tal critério define que uma política π_1 é preferida em relação à π_2 se o custo esperado superior de π_1 for estritamente menor que o custo esperado inferior de π_2 e π_1 é considerada maximal se não existir uma política que seja preferida no lugar de π_1.

Esse critério de preferência e maximalidade proposto em [Harmanec, 1999] gera um conjunto parcialmente ordenado de políticas maximais, pois algumas políticas não serão comparáveis. Apesar de não fazer nenhuma suposição sobre o comportamento da natureza na escolha de eventos incertos, característica de principal interesse do trabalho, esse critério se torna computacionalmente inviável para MDPIPs gerais. Isso porque o algoritmo de solução deve considerar todas as políticas potencialmente maximais durante a sua busca, que em uma análise superficial representa $O(|\mathcal{A}|^{|\mathcal{S}|})$ políticas. Como apontado pelo próprio autor, ainda é necessário realizar estudos para encontrar casos especiais MDPIPs onde esses critérios possam ser usados de forma eficientes.

Capítulo 5

Conclusão

Todo o negócio presegue baseado em crenças, ou jugamento de probabilidades, e não em certezas.

— Charles W. Eliot, 1834–1934
(químico norte-americano)[1]

Nesse trabalho foram apresentadas as duas abordagens usadas na área de planejamento em IA para resolver problemas que envolvam incerteza: planejamento não-determinístico e planejamento probabilístico. Essas duas abordagens são aplicas em problemas nos quais o efeito das ações está associado, respectivamente, à incerteza knightiana e risco.

Também foi exibido que existe uma família de problemas, chamada de problemas de planejamento sob incerteza, em que tanto incerteza knightiana quanto risco co-existem. Existem muitas razões para as quais esse tipo de problema pode surgir, como por exemplo: (i) a falta de informação sobre o ambiente em que o agente está imerso [Levi, 1980; Walley, 1991; 1996]; (ii) um grupo de especialistas podem discordar sobre os valores de probabilidades e apenas a coleção da opinião de cada um pode ser considerado uma decisão válida [Seidenfeld *et al.*, 1989; Seidenfeld e Schervish, 1990], e (iii) quando o interesse é a robustez das inferências, i.e., na avaliação de quanto variam as inferências quando é permitido que o valor das probabilidades variem [Berger, 1985; Huber, 1980; Kadane, 1984]

Para os problemas de planejamento sob incerteza, as soluções obtidas pelas abordagens atuais de consistem em assumir apenas uma forma de manifestação de incerteza. Na solução não-determinística o risco associado ao efeito das ações é transformado em incerteza knightiana, gerando perda de informação. Já a solução probabilísticas substitui incerteza knightiana por risco. Isso é feito distribuindo de forma igual a massa de probabilidade entre cada efeito possível, o que modificar o problema original.

Ao longo desse trabalho foi definido um modelo, os processos markovianos de decisão com transição valorada por conjunto (MDPSTs), capaz de expressar problemas de planejamento sob incerteza sem perda de informações ou geração de restrições adicionais. Esse modelo é uma generalização dos modelos propostos para planejamento não-determinístico e planejamento probabilístico, pois contém como caso especial o modelo dos processos markovianos de decisão (MDPs) e o modelo não-determinístico de transição de estados.

Além de definir formalmente os MDPSTs, também foi exibido que qualquer problema descrito como um MDPST pode ser convertido em um MDP com probabilidades imprecisas (MDPIP). Através dessa redução (Proposição 1), dois algoritmos da área de pesquisa operacional, iteração de valor e iteração de política, foram adaptados para MDPSTs, bem como dois dos principais algoritmos da área de planejamento, o RTDP (*Real Time Dynamic Programming*) e o LRTDP (*Labelled RTDP*) [Bonet e Geffner, 2003].

Também foi exibido o princípio ótimo de Bellman para MDPSTs (Teorema 3), que é uma simplificação do mesmo princípio para MDPIPs. Essa simplificação foi obtida devido à diferença de expressividade entre

[1] Charles W. Eliot foi presidente da universidade de Harvard por 40 anos (1869–1909), período no qual ele transformou Harvard em uma das mais importantes universidades da atualidade.

MDPIPs e MDPSTs (Proposição 1). Através da análise de complexidade para o pior caso do cálculo do princípio ótimo de Bellman, foi possível verificar que resolver um problema de planejamento probabilístico usando um MDPST possui a mesma ordem de complexidade que resolvê-lo com um MDP. Essa resultado implica que o custo computacional pago por usar os MDPSTs, que são mais gerais, está associado apenas com *quantidade* de incerteza do problema e não ao modelo em si.

5.1 Principais contribuições

A principal contribuição desse trabalho é a caracterização e formalização da família de problemas de planejamento sob incerteza, que é negligenciado pela comunidade de planejamento. Outra contribuição que vale destacar é o Teorema 3, no qual é provado que o princípio ótimo de Bellman para MDPIPs pode ser simplificado para os MDPSTs. Assim, dado um MDPST $q = \langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$, ao invés de convertê-lo para um MDPIP $r = \langle \mathcal{S}, P_0, \mathcal{A}, \mathcal{K}, C \rangle$ e depois calcular o princípio ótimo de Bellman para MDPIPs, é possível calcular esse valor diretamente em q. Em termos de complexidade computacional no pior caso, isso representa que, ao invés de cada iteração ter custo $O(|\mathcal{S}|^P |\mathcal{A}| \overline{F}^Q)$ para $P \geq 2$ e $Q \geq 1$, ela custará apenas $O(|\mathcal{S}|^2 |\mathcal{A}| \overline{F})$. Esse simplificação é possível, por que ao invés de cada iteração consistir em resolver um sistema linear como nos MDPIPs, cada iteração é reduzida a um problema da mochila fracionária, que possui uma solução gulosa ótima. Outras contribuições importantes são:

- O desenvolvimento do modelo dos MDPSTs (MDPST1 – MDPST5, capaz de expressar as dinâmicas básicas de ações e a dinâmica baseada em incerteza.
- A Proposição 1, na qual é provado, de forma construtiva, que qualquer problema representável por um MDPST pode ser representado por um MDPIP. Essa proposição faz o elo entre o modelo proposto e trabalhos da área de pesquisa operacional [Satia e Lave Jr, 1973; White III e Eldeib, 1994; Harmanec, 1999].
- A adaptação, através da Proposição 1 e do Teorema 3, dos seguintes algoritmos clássicos da área de pesquisa operacional baseados em enumeração de estados: iteração de valor e iteração de política.
- A Proposição 4, na qual é demonstrada uma condição suficiente para garantir que um SPST, um caso particular de MDPST, respeita a hipótese de alcançabilidade. Essa hipótese é necessária para garantir que os algoritmos RTDP e LRTDP não entrem em um ciclo infinito *loop* e devolvam a solução ótima.
- O acréscimo à literatura em português de uma revisão sobre os modelos usados em planejamento e dos principais algoritmos para planejamento probabilístico.

Como resultado preliminar desse estudo, foi publicado o artigo [Trevizan *et al.*, 2006], no qual é formalizado o problema de planejamento sob incerteza e, como resultado, são fornecidos apenas a Proposição 1 e o algoritmo ITERAÇÃO-DE-POLÍTICA-SATIA. Todos os resultados obtidos nesse estudo foram submetidos para o *International Joint Conference on Artificial Intelligence* [Trevizan *et al.*, 2007].

5.2 Trabalhos futuros

Entre as diversas extensões possíveis para esse trabalho, *uso de MDPSTs para aprendizado por reforço* e *análise de sensibilidade* são duas extensões especialmente interessantes.

5.2.1 Uso de MDPSTs para aprendizado por reforço

O objetivo da área de aprendizado por reforço (AR) [Sutton e Barto, 1998] pode ser visto como resolver um problema de decisão seqüencial sob risco, geralmente um MDP, onde probabilidade de transição de estados $P(\cdot|s,a)$ não são conhecidas. Nesse tipo problema o agente decisor busca por uma política ótima ao mesmo tempo que estima os valores de $P(\cdot|s,a)$.

Alguns trabalhos de AR, como [Szepesvari e Littman, 1996], apresentam generalizações de MDPs, porém, esses modelos gerais não propõem mudanças no tipo de incerteza modelado. Por isso, os algoritmos de AR sempre mantém uma estimativa, em alguns casos exata, para $P(\cdot|s,a)$.

Uma extensão desse trabalho seria usar MDPSTs para resolver problemas da área de AR. A abordagem usando MDPSTs permitiria um transição suave entre o cenário inicial de não-determinismo e o cenário de risco obtido quando é os valores de probabilidade convergiram. Dessa forma, as funções $\mathbf{F}(s,a)$ e $m(\cdot|s,a)$ de um MDPST $q\langle \mathcal{S}, P_0, \mathcal{A}, m, C\rangle$ seriam inicialmente definidas por: $\forall s \in \mathcal{S}, a \in \mathcal{A}(s)$: $\mathbf{F}(s,a) = \{\mathcal{S}\}$ e $m(\{\mathcal{S}\}|s,a) = 1$. Ao longo do tempo, um conjunto de estados possíveis $k \in \mathbf{F}(s,a)$ seria particionado em $k_1, \ldots, k_n$ e a massa de probabilidade $m(k|s,a)$ seria dividida, não necessariamente de maneira uniforma, entre $k_1, \ldots, k_n$. Um critério de convergência para a medida de probabilidade $P(\cdot|s,a)$ poderia ser definido como: se existir apenas um conjunto de estados possíveis $k \in F(s,a)$ tal que $|k| > 1$ e $m(k|s,a) \leq \epsilon \ll 1$ então $m(\cdot|s,a)$ convergiu para $P(\cdot|s,a)$.

5.2.2 Análise de sensibilidade

A análise de sensibilidade pode ser vista, de maneira simplista, como o estudo do comportamento de uma variável aleatória ao variar sua medida de probabilidade associada. Em um MDPST, seria possível definir um novo critério de decisão, no qual uma política π seria ótima se $\overline{E}[\pi] - \underline{E}[\pi]$ for mínimo.

Outro critério possível é minimizar o *custo médio por decisão* esperado [Kalyanasundaram *et al.*, 2002]. O custo médio por decisão é definido por (5.1), e representa uma forma alternativa à função valor (2.2) para avaliar históricos de horizonte infinito. Dessa forma, $E_{P(\cdot|\pi)}[G] = P_0(s^0) \sum_{h \in \mathcal{H}} G(h) P(h|\pi)$ e uma política π será ótima se ela minimizar $E_{P(\cdot|\pi)}[G]$.

$$G(h = \langle s^0, a^0, \ldots, s^n\rangle) = \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n} C(s^i, a^i) \tag{5.1}$$

Apêndice A

Prova da simplificação da equação de Bellman para MDPSTs

Antes de provar o Teorema 3 é necessário fazer algumas definições (Definição 3 e 4) bem como uma nova proposição (Proposição 5).

Definição 3. *A medida de probabilidade induzida pelo princípio ótimo de Bellman para MDPSTs* (3.9)*, representada por $P_{st}(\cdot)$, é definida para todo $s, s' \in \mathcal{S}$ e para todo $a \in \mathcal{A}(s)$ por:*

$$P_{st}(s'|s,a) = \sum_{\substack{k \in \mathbf{F}(s,a) \\ s' = \text{argmax}_{s' \in k} V(s')}} m(k|s,a)$$

Dado um MDPST $q = \langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$ e o seu MDPIP associado $r = \langle \mathcal{S}, P_0, \mathcal{A}, \mathcal{K}, C \rangle$, a medida de probabilidade $P_{st}(\cdot)$ de q pertence ao conjunto credal $\mathcal{K}(\cdot)$ de r. Isso porque, por construção, $P_{st}(\cdot)$ respeita a Proposição 1, ou seja, é uma distribuição válida das massas de probabilidade de q.

Usando a Definição 3, é possível reescrever o princípio ótimo de Bellman para MDPSTs (3.9) como:

$$V^*(s) = \min_{a \in \mathcal{A}(s)} \{C(s,a) + \gamma \sum_{s' \in \mathcal{S}} P_{st}(s'|s,a) V^*(s')\}$$

Definição 4. *Dado um MDPST $q = \langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$, seu MDPIP associado $r = \langle \mathcal{S}, P_0, \mathcal{A}, \mathcal{K}, C \rangle$ e uma medida de probabilidade $P(\cdot) \in \mathcal{K}(\cdot)$, a função $fm(\cdot, \cdot, P)$ associa, para cada $s \in \mathcal{S}, a \in \mathcal{A}(s), k \in \mathbf{F}(s,a)$, e $s' \in k$, a parte da massa $m(k|s,a)$ associada ao estado s' pela medida de probabilidade P. Em outras palavras, $fm(\cdot, k, P)$ exibe como a massa $m(k|s,a)$ foi distribuída por $P(\cdot)$ entre os estados $s' \in k$.*

Note que pela Definição 4 a seguinte igualdade, para todos os estados $s, s' \in \mathcal{S}$, ação $a \in \mathcal{A}(s)$ e $P(\cdot) \in \mathcal{K}(\cdot)$, vale:

$$P(s'|s,a) = \sum_{\substack{k \in \mathbf{F}(s,a) \\ s' \in k}} fm(s', k, P(\cdot|s,a))$$

Proposição 5. *Dado um MDPST $q = \langle \mathcal{S}, P_0, \mathcal{A}, m, C \rangle$ e uma função valor $V \colon \mathcal{S} \to \mathbb{R}_+$, a seguinte relação entre q e o seu MDPIP associado $r = \langle \mathcal{S}, P_0, \mathcal{A}, \mathcal{K}, C \rangle$ é válida para todo $s \in \mathcal{S}$ e para todo $a \in \mathcal{A}(s)$:*

$$\max_{P_{ip} \in \mathcal{K}} E_{P_{ip}}[V] = E_{P_{st}}[V]. \tag{A.1}$$

Demonstração. Essa prova será dividida em dois lemas, onde cada um consiste em demonstrar que um lado de (A.1) é maior ou igual que o outro.

Lema 6. $\max_{P_{ip}\in\mathcal{K}} E_{P_{ip}}[V] \geq E_{P_{st}}[V]$.

Demonstração. Através da Proposição 1, $P_{st}(\cdot|s,a) \in \mathcal{K}$ por construção, logo essa inequação é válida. □

Lema 7. $\max_{P_{ip}\in\mathcal{K}} E_{P_{ip}}[V] \leq E_{P_{st}}[V]$.

Demonstração. Essa prova será feita por contradição assumindo que (A.2) vale.

$$\max_{P_{ip}\in\mathcal{K}} E_{P_{ip}}[V] > E_{P_{st}}[V] \tag{A.2}$$

Através dessa suposição, é possível deduzir que existe $\overline{k} \in \mathbf{F}$ tal que $fm(\text{argmax}_{s'\in\overline{k}} V(s'), \overline{k}, P_{ip}) < m(\overline{k}|s,a)$, caso contrário, para todo $k' \in \mathbf{F}(s,a)$, $fm(\text{argmax}_{s'\in k'} V(s'), k', P_{ip}) = m(k'|s,a)$, logo $P_{ip}(s'|s,a) = P_{st}(s'|s,a)$, o que é exatamente o objetivo dessa prova.

Seja $\overline{s} = \text{argmax}_{s'\in\overline{k}} V(s')$ e a medida de probabilidade $\overline{P}(\cdot|s,a)$ definida por:

$$\overline{P}(s'|s,a) = \begin{cases} P_{ip}(\overline{s}|s,a) - fm(\overline{s},\overline{k},P_{ip}) + m(\overline{k}|s,a) & \text{se } s' = \overline{s}, \\ P_{ip}(s'|s,a) - fm(s',\overline{k},P_{ip}) & \text{se } s' \neq \overline{s} \text{ e } s' \in \overline{k}, e \\ P_{ip}(s'|s,a) & \text{caso contrário} \end{cases}$$

Note que $\overline{P}(\cdot|s,a) \in \mathcal{K}(\cdot|s,a)$, pois a única alteração feita sobre $P_{ip}(\cdot|s,a)$ consiste em movimentar a massa do conjunto de estados alcançáveis $\overline{k}$ entre os estados contidos em $\overline{k}$, o que não viola nenhuma das restrições do modelo. Note que $\overline{P}(\cdot|s,a)$ gera uma contradição, pois:

$$\begin{aligned} E_{\overline{P}}[V] - \max_{P_{ip}\in\mathcal{K}} E_{P_{ip}}[V] &= \sum_{s'\in\mathcal{S}} V(s')(\overline{P}(s'|s,a) - P_{ip}(s'|s,a)) \\ &= V(\overline{s})m(\overline{k}|s,a) - \sum_{s'\in\overline{k}} V(s')fm(s',\overline{k},P_{ip}) \\ &> 0. \end{aligned}$$

Logo $\max_{P_{ip}\in\mathcal{K}} E_{P_{ip}}[V] \leq E_{P_{st}}[V]$ vale. □

Através do Lema 6 e do Lema 7, a relação descrita por essa proposição (A.1) também é verdadeira. □

Teorema 3. *Para qualquer MDPST $q = \langle \mathcal{S}, P_0, \mathcal{A}, m, C\rangle$ e seu MDPIP associado $r = \langle \mathcal{S}, P_0, \mathcal{A}, \mathcal{K}, C\rangle$, o princípio ótimo de Bellman para MDPIPs* (3.3) *é equivalente à:*

$$V^*(s) = \min_{a\in\mathcal{A}(s)} \{C(s,a) + \gamma \sum_{k\in\mathbf{F}(s,a)} m(k|s,a) \max_{s'\in k} V^*(s')\} \tag{A.3}$$

Demonstração. Note que o Teorema 3 pode ser reformulado como: para todo $s \in \mathcal{S}$, a seguinte relação

$$V^*_{ip}(s) = V^*_{st}(s), \tag{A.4}$$

vale, onde $V^*_{ip}(s)$ é o princípio ótimo de Bellman para MDPIPs (3.3) e $V^*_{st}(s)$ é o princípio ótimo de Bellman para MDPSTs proposto nesse trabalho (3.9).

Para atingir o ponto fixo de ambas as recorrências é necessário criar uma seqüência convergente para cada uma delas: $\langle V^0_{ip}, V^1_{ip}, \ldots, V^n_{ip}, \ldots, V^*_{ip}\rangle$ e $\langle V^0_{st}, V^1_{st}, \ldots, V^n_{st}, \ldots, V^*_{st}\rangle$. É necessário provar que, para todo $s \in \mathcal{S}$, as duas seqüências são iguais. Para gerar tais seqüências, será usado o princípio ótimo de Bellman para programação dinâmica:

$$V_{ip}^{t+1}(s) = \min_{a\in\mathcal{A}(s)} \max_{P\in\mathcal{K}} \{C(s,a) + \gamma \sum_{s'\in F(s,a)} P(s'|s,a) V_{ip}^{t}(s')\} \tag{A.5}$$

$$V_{st}^{t+1}(s) = \min_{a\in\mathcal{A}(s)} \{C(s,a) + \gamma \sum_{k\in\mathbf{F}(s,a)} m(k|s,a) \max_{s'\in k} V_{st}^{t}(s')\} \tag{A.6}$$

Essa prova será feita por indução, e como base será usado o fato que $V_{ip}^{1}(s) = V_{st}^{1}(s)$. A base dessa indução é verdadeira porque, por definição, $V_{ip}^{0}(s) = V_{st}^{0}(s) = 0$, logo, $V_{ip}^{1}(s) = \min_{a\in\mathcal{A}(s)} C(s,a) = V_{st}^{1}(s)$.

Como passo da indução, suponha que $V_{ip}^{n}(s) = V_{st}^{n}(s)$. Deseja-se provar que $V_{ip}^{n+1}(s) = V_{st}^{n+1}(s)$, e isso será feito provando que, para todo $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$, (A.7) vale. Caso isso seja verdade, então conjunto de escolha do operador $\min_{a\in\mathcal{A}(s)}$ em (A.5) e (A.6) será o mesmo, logo o resultado desse operador também será o mesmo.

$$\max_{P\in\mathcal{K}} \sum_{s'\in F(s,a)} P(s'|s,a) V_{ip}^{n}(s') = \sum_{k\in\mathbf{F}(s,a)} m(k|s,a) \max_{s'\in k} V_{st}^{n}(s'). \tag{A.7}$$

A equação (A.7) pode ser reescrita usando a Definição 3 e a variável aleatória $V^n : \mathcal{S} \to \mathbb{R}_+$, onde para todo $s \in \mathcal{S}, V^n(s) = V_{ip}^{n}(s) = V_{st}^{n}(s)$, por:

$$\max_{P\in\mathcal{K}} E_P[V^n] = E_{P_{st}}[V^n]. \tag{A.8}$$

A Proposição 5 garante que (A.8) vale para todo $s \in \mathcal{S}$ e $a \in \mathcal{A}(s)$, logo, para todo $s \in \mathcal{S}$, $V_{ip}^{n+1}(s) = V_{st}^{n+1}(s)$ também vale. Isso prova o passo da indução, o que garante que as seqüências $\langle V_{ip}^{0}, V_{ip}^{1}, \ldots, V_{ip}^{n}, \ldots, V_{ip}^{*}\rangle$ e $\langle V_{st}^{0}, V_{st}^{1}, \ldots, V_{st}^{n}, \ldots, V_{st}^{*}\rangle$ são iguais, ou seja, elas convergem para o mesmo ponto fixo. □

Índice Remissivo

Referências Bibliográficas

[Barto *et al.*, 1995] A.G. Barto, S.J. Bradtke, e S.P. Singh. Learning to act using real-time dynamic programming. *Artificial Intelligence*, 72(1-2):81–138, 1995.

[Bellman, 1957] Richard E. Bellman. *Dynamic Programming*. Princeton University Press, Princeton, New Jersey, 1957.

[Berger, 1985] J.O. Berger. *Statistical Decision Theory and Bayesian Analysis*. Springer-Verlag, 1985.

[Berman e Plemmons, 1979] A. Berman e R.J. Plemmons. *Nonnegative matrices in the mathematical sciences*. Academic Press, NY, 1979.

[Bertoli *et al.*, 2001] P. Bertoli, A. Cimatti, M. Roveri, e P. Traverso. Planning in nondeterministic domains under partial observability via symbolic model checking. Em *Proc. of the 17th International Joint Conference on Artificial Intelligence*, páginas 473–478, 2001.

[Bertsekas e Tsitsiklis, 1991] D. P. Bertsekas e J. N. Tsitsiklis. An analysis of stochastic shortest path problems. *Mathemathics of Operations Research*, 16:580–595, 1991.

[Bonet e Geffner, 1999] B. Bonet e H. Geffner. Planning as heuristic search: New results. Em S. Biundo e M. Fox, editores, *Proc. 5th European Conference on Planning*, páginas 359–371, Durham, UK, 1999. Springer: Lecture Notes on Computer Science.

[Bonet e Geffner, 2000] B. Bonet e H. Geffner. Planning with incomplete information as heuristic search in belief space. Em S. Chien, S. Kambhampati, e C. Knoblock, editores, *Proc. of the 6th International Conference on Artificial Intelligence Planning and Scheduling*, páginas 52–61, Breckenridge, CO, 2000. AAAI Press.

[Bonet e Geffner, 2001a] B. Bonet e H. Geffner. GPT: a tool for planning with uncertainty and partial information. Em A. Cimatti, H. Geffner, E. Giunchiglia, e J. Rintanen, editores, *Proc. of the 17th International Joint Conference on Artificial Inelligence: Workshop on Planning with Uncertainty and Partial Information*, páginas 82–87, Seattle, WA, 2001.

[Bonet e Geffner, 2001b] B. Bonet e H. Geffner. Heuristic search planner 2.0. *AI Magazine*, 22(3):77–80, Fall 2001.

[Bonet e Geffner, 2001c] B. Bonet e H. Geffner. Planning and control in artificial intelligence: A unifying perspective. *Applied Intelligence*, 14(3):237–252, 2001.

[Bonet e Geffner, 2001d] B. Bonet e H. Geffner. Planning as heuristic search. *Artificial Intelligence*, 129(1–2):5–33, 2001.

[Bonet e Geffner, 2003] B. Bonet e H. Geffner. Labeled RTDP: Improving the convergence of real-time dynamic programming. Em E. Giunchiglia, N. Muscettola, e D. Nau, editores, *Proc. 13th International Conference on Automated Planning and Scheduling*, páginas 12–21, Trento, Italy, 2003. AAAI Press.

[Bonet e Geffner, 2006] B. Bonet e H. Geffner. Learning Depth-First Search: A unified approach to heuristic search in deterministic and non-deterministic settings, and its application to MDPs. Em *Proc. 16th International Conference on Artificial Intelligence Planning and Scheduling*, páginas 142–151, 2006.

[Boutilier *et al.*, 1999] C. Boutilier, T. Dean, e S. Hanks. Decision-Theoretic Planning: Structural Assumptions and Computational Leverage. *Journal of Artificial Intelligence Research*, 11:1–94, 1999.

[Cheeseman, 1985] P. Cheeseman. In defense of probability. Em *Proc. of the 9th International Joint Conference on Artificial Intelligence*, página 1002, 1985.

[Cozman, 1997] F. G. Cozman. A brief introduction to the theory of sets of probability measures. Relatório técnico CMU-RI-TR 97-24, School of Computer Science, Carnegie Mellon University, 1997.

[Cozman, 2005a] F. G. Cozman. Probability theory in artificial intelligence. Apostila curso de teoria de probabilidades em inteligência artificial e robótica, Escola Politécnica, Universidade de São Paulo, 2005.

[Cozman, 2005b] F.G. Cozman. Graphical models for imprecise probabilities. *International Journal of approximate reasoning*, 39(2-3):167–184, 2005.

[Dean *et al.*, 1997] T. Dean, R. Givan, e S. Leach. Model reduction techniques for computing approximately optimal solutions for markov decision processes. Em *Proc. of the 13th Conference on Uncertainty in Artificial Intelligence*, páginas 124–131, New York, NY, 1997. Elsevier Science Publishing Comapny, Inc.

[Fox e Long, 2003] M. Fox e D. Long. PDDL2.1: An extension to PDDL for expressing temporal planning domains. *Journal of Artificial Intelligence Research*, 20:61–124, 2003.

[Geffner, 2002] Héctor Geffner. Perspectives on artificial intelligence planning. Em *Proc. of the 18th national conference on Artificial intelligence*, páginas 1013–1023, Menlo Park, CA, USA, 2002. American Association for Artificial Intelligence.

[Ghallab *et al.*, 2004] M. Ghallab, D. Nau, e P. Traverso. *Automated Planning: Theory & Practice*. Morgan Kaufman, 2004.

[Giron e Rios, 1980] F.J. Giron e S. Rios. Quasi-bayesian behaviour: A more realistic approach to decision making? *Bayesian Statistics*, páginas 17–38, 1980.

[Givan *et al.*, 2000] R. Givan, S. M. Leach, e T. Dean. Bounded-parameter Markov Decision Processes. *Artificial Intelligence*, 122(1-2):71–109, 2000.

[Givan *et al.*, 2003] R. Givan, T. Dean, e M. Greig. Equivalence notions and model minimization in markov decision processes. *Artificial Intelligence*, 147(1-2):163–223, 2003.

[Greenspan, 2004] A. Greenspan. Risk and uncertainty in monetary policy. *The American Economic Review*, 94(2):33–40, 2004.

[Harmanec, 1999] D. Harmanec. A generalization of the concept of markov decision process to imprecise probabilities. Em *Proc. of the 1st International Symposium On Imprecise Probability: Theories And Applications*, páginas 175–182, 1999.

[Hoffmann e Nebel, 2001] J. Hoffmann e B. Nebel. The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, 14:253–302, 2001.

[Howard, 1960] R. A. Howard. *Dynamic Porgramming and Markov Processes*. MIT Press, 1960.

[Huber, 1980] P.J. Huber. *Robust Statistics*. Wiley, New York, 1980.

[Kadane, 1984] J.B. Kadane. *Robustness of Bayesian Analysis.*, volume 4. Elsevier Science Pub. Co., New York, 1984.

[Kalyanasundaram *et al.*, 2002] S. Kalyanasundaram, E.K.P. Chong, e N.B. Shroff. Markov decision processes with uncertain transition rates: sensitivity and robust control. Em *Proc. of the 41st IEEE Conference on Decision and Control*, volume 4, páginas 3799–3804, 2002.

[Karmarkar, 1984] N. Karmarkar. A new polynomial-time algorithm for linear programming. *Combinatorica*, 4(4):373–395, 1984.

[Kautz e Selman, 1999] H. Kautz e B. Selman. Unifying SAT-based and graph-based planning. Em Jack Minker, editor, *Workshop on Logic-Based Artificial Intelligence*, College Park, Maryland, 06 1999. Computer Science Department, University of Maryland.

[Knight, 1921] F.H. Knight. *Risk, Uncertainty, and Profit.* Hart, Schaffner & Marx; Houghton Mifflin Company, Boston, 1921.

[Kojima *et al.*, 1988] M. Kojima, S. Mizuno, e A. Yoshise. A primal-dual interior point algorithm for linear programming. *Progress in Mathematical Programming Interior-point and related methods table of contents*, páginas 29–47, 1988.

[Levi, 1980] I. Levi. *The Enterprise of Knowledge.* MIT Press, 1980.

[Lewis e Papadimitriou, 1997] Harry R. Lewis e Christos H. Papadimitriou. *Elements of the Theory of Computation.* Prentice Hall, 1997.

[Luce e Raiffa, 1957] D. Luce e H. Raiffa. *Games and Decisions.* Dover edition, Mineola, 1957.

[McDermott *et al.*, 1998] D. McDermott, M. Ghallab, A. Howe, A. Ram, M. Veloso, D. S. Weld, e D. E. Wilkins. PDDL - the planning domain definition language. Relatório técnico CVC TR-98-003/DCS TR-1165, Yale Center for Computational Vision and Control, 1998.

[Nau *et al.*, 1999] D. Nau, Y. Cao, A. Lotem, e H. Munoz-Avila. SHOP: Simple hierarchical ordered planner. Em T. Dean, editor, *Proc. of the 16th International Joint Conference on Artificial Intelligence*, páginas 968–975, Stockholm, SE, Agosto 1999. Morgan Kaufmann Publishers Inc.

[Nau *et al.*, 2003] D. Nau, T.C. Au, O. Ilghami, U. Kuter, W. Murdock, D. Wu, e F. Yaman. SHOP2: An HTN planning system. *Journal of Artificial Intelligence Research*, 20:379–404, december 2003.

[Newell e Simon, 1963] A. Newell e H. Simon. GPS: A program that simulates human thought. Em E. A. Feigenbaum e J. Feldman, editores, *Computers and Thought.* McGraw-Hill, New York, 1963.

[Papadimitriou, 1994] Christos H. Papadimitriou. *Computational Complexity.* Addison-Wesley, 1994.

[Penberthy e Weld, 1992] J. Scott Penberthy e Daniel S. Weld. UCPOP: A sound, complete, partial order planner for ADL. Em *Principles of Knowledge Representation and Reasoning: Proc. of the 3rd International Conference (KR-92)*, páginas 103–114, Cambridge, MA, Outubro 1992. Morgan Kaufmann.

[Pereira e Barros, 2004] S. L. Pereira e L. N. Barros. Formalizing planning algorithms: a logical framework for the research on extending the classical planning approach. Em *Proc. of the 14th International Conference on Automated Planning & Scheduling: Workshop on Connecting Planning Theory with Practice*, 2004.

[Perny *et al.*, 2005] P. Perny, O. Spanjaard, e P. Weng. Algebraic markov decision processes. Em *Proc. of the 19th International Conference on Artificial Intelligence*, páginas 1372–1377, 2005.

[Puterman e Shin, 1978] M. L. Puterman e M. C. Shin. Modified policy iteration algorithms for discounted markov decision problems. *Management Science*, 24:1127–1137, 1978.

[Puterman, 1994] M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* John Wiley & Sons, New York, 1994.

[Russel e Norvig, 2003] S. Russel e P. Norvig. *Artificial Intelligence: A Modern Approach*. Prentice-Hall, Inc., 2nd. edição, 2003.

[Satia e Lave Jr, 1973] J. K. Satia e R.E. Lave Jr. Markovian decision processes with uncertain transition probabilities. *Operations Research*, 21(3):728–740, 1973.

[Seidenfeld *et al.*, 1989] T. Seidenfeld, J.B. Kadane, e M.J. Schervish. On the shared preferences of two Bayesian decision makers. *The Journal of Philosophy*, 86(5):225–244, 1989.

[Seidenfeld e Schervish, 1990] T. Seidenfeld e M. Schervish. Two perspectives on consensus for (Bayesian) inference and decisions. *IEEE Transactions on Systems, Man and Cybernetics*, 20(1):318–325, 1990.

[Shafer, 1976] G. Shafer. *A Mathematical Theory of Evidence*. Princeton University Press, 1976.

[Shapley, 1953] L.S. Shapley. Stochastic games. Em *Proc. of the National Academy of Sciences of the United States of America*, volume 39, páginas 1095–1100, 1953.

[Smith *et al.*, 1996] S. J. J. Smith, D. S. Nau, e T. A. Throop. Total-order multi-agent task-network planning for contract bridge. Em *Proc. of the 13th National Conference on Artificial Intelligence*, páginas 108–113. AAAI Press / MIT Press, 1996.

[Smith *et al.*, 1998] S.J.J. Smith, D.S. Nau, e T. Throop. Success in spades: Using AI planning techniques to win the world championship of computer bridge. Em *Proc. of the 15th National Conference on Artificial Intelligence and of the 10th Conference on Innovative Applications of Artificial Intelligence*, páginas 1079–1086, Menlo Park, 1998. AAAI Press.

[Sutton e Barto, 1998] R.S. Sutton e A.G. Barto. *Reinforcement learning: an introduction*. MIT Press, 1998.

[Szepesvari e Littman, 1996] C. Szepesvari e M.L. Littman. Generalized markov decision processes: Dynamic-programming and reinforcement-learning algorithms. Em *Proc. of the 13th International Conference of Machine Learning*, Bari, 1996.

[Trevizan *et al.*, 2006] F. W. Trevizan, F. G. Cozman, e L. N. de Barros. Unifying Nondeterministic and Probabilistic Planning through Imprecise Markov Decision Processes. Em *Proc. the 10th Ibero-American Conference on AI (IBERAMIA) and 18th Brazilian AI Symposium (SBIA)*, volume 4140 de *Lecture Notes in Computer Science*, páginas 502–511, Ribeirão Preto, SP, Brazil, 10 2006. Springer-Verlag. (Winner of the Best Paper Award).

[Trevizan *et al.*, 2007] F. W. Trevizan, F. G. Cozman, e L. N. de Barros. Planning under risk and knightian uncertainty. Em *Proc. of the 20th International Joint Conference on Artificial Inteligence*. (submetido), 2007.

[Walley, 1991] P. Walley. *Statistical Reasoning with Imprecise Probabilities*. Chapman and Hall, London, 1991.

[Walley, 1996] P. Walley. Measures of uncertainty in expert systems. *Artificial Intelligence*, 83:1–58, 1996.

[Watkins e Dayan, 1992] C. J. Watkins e P. Dayan. Q-learning. *Machine Learning*, 8(3/4):279–292, 1992.

[White III e Eldeib, 1994] Chelsea C. White III e Hany K. Eldeib. Markov decision processes with imprecise transition probabilities. *Operations Research*, 42(4):739–749, 1994.