

Cassio Polpo de Campos

*Redes Credais e Qualitativas:
Complexidade e Algoritmos*

Tese apresentada à Escola Politécnica
da Universidade de São Paulo para
obtenção do título de Doutor em
Engenharia.

São Paulo
2005

Cassio Polpo de Campos

PARA CÓPIAS, CONSULTAR A EDIÇÃO REVISADA :

FT- 2168

Ed. rev.

***Redes Credais e Qualitativas:
Complexidade e Algoritmos***

Tese apresentada à Escola Politécnica
da Universidade de São Paulo para
obtenção do título de Doutor em
Engenharia.

Área de Concentração: Engenharia
Mecatrônica

Orientador: Prof. Dr. Fabio Gagliardi Cozman

São Paulo
2005

À minha família

Resumo

Uma rede credal é uma ferramenta gráfica para representação e manipulação de incerteza, onde valores de probabilidade podem ser imprecisos ou indeterminados. Uma rede credal associa um grafo direcionado acíclico a uma coleção de conjuntos de medidas de probabilidade. Neste contexto, uma inferência é o cálculo de valores máximos e mínimos para probabilidades condicionais. Neste trabalho apresentamos resultados de complexidade de problemas de inferência e novas técnicas para inferência em redes credais baseados em programação multilinear. Através dessas idéias desenvolvemos novos algoritmos exatos e aproximados para inferência em tais redes. Experimentos indicam que estes algoritmos tem melhor desempenho que os conhecidos até o momento no sentido de produzir resultados mais precisos para redes de grande porte. Tratamos ainda outros modelos gráficos, como redes qualitativas e semi-qualitativas, pois elas também se encaixam no contexto das redes credais.

Palavras-chave: Inferência com incerteza. Conjuntos de medidas de probabilidade. Redes Bayesianas e qualitativas. Complexidade. Programação não linear e multilinear. Algoritmos exatos e aproximados.

Abstract

A credal network is a graphical tool for representation and manipulation of uncertainty, where probability values may be imprecise or indeterminate. A credal network associates a directed acyclic graph with a collection of sets of probability measures; in this context, inference is the computation of tight lower and upper bounds for conditional probabilities. In this work we present complexity results for the inference problems and new techniques for inference in credal networks based on multilinear programming. These techniques have led to new exact and approximate algorithms for inference in such networks. Experiments indicate that these new algorithms have better performance than existing ones, in the sense that they produce more accurate results in larger networks. We also deal with other graphical models as qualitative and semi-qualitative networks, as they also fit in the credal network model.

Keywords: Uncertainty and reasoning. Sets of probability measures. Bayesian and qualitative networks. Complexity. Nonlinear and multilinear programming. Exact and approximating algorithms.

Agradecimentos

Agradeço a todas as pessoas envolvidas diretamente ou indiretamente na elaboração deste trabalho. Neste sentido, cito alguns nomes fundamentais, sem prejuízo dos demais: Fabio Gagliardi Cozman pela constante e competente orientação, Aritanan Borges Garcia Gruber pelos comentários e dicas, Edith Ranzini pelo apoio na Escola Politécnica, colegas do Laboratório de Tomada de Decisão pelo compartilhamento de experiências e pesquisas, e minha família pela ajuda e paciência.

Sumário

Lista de Figuras

Lista de Tabelas

1 Preliminares	p. 11
1.1 Introdução e objetivo	p. 11
1.2 Visão geral do documento	p. 12
2 Modelos gráficos para inferência	p. 14
2.1 Introdução	p. 14
2.2 Redes Bayesianas	p. 15
2.3 Conjuntos e redes credais	p. 17
2.3.1 Conjuntos credais	p. 17
2.3.2 Redes credais	p. 18
2.3.3 Conceitos de independência	p. 19
2.4 Redes qualitativas e semi-qualitativas	p. 20
2.4.1 Redes qualitativas	p. 20
2.4.2 Redes semi-qualitativas	p. 23

2.5	Redes de acordo com a topologia	p. 24
2.6	Inferência na presença de imprecisão	p. 26
3	Resultados de complexidade	p. 29
3.1	Introdução	p. 29
3.2	Redes Bayesianas	p. 30
3.2.1	Atualização de crenças	p. 30
3.2.2	Explicação mais provável	p. 31
3.2.3	Máximo a posteriori	p. 33
3.2.4	Maximin a posteriori	p. 34
3.3	Redes credais	p. 37
3.3.1	Atualização de crenças	p. 37
3.3.2	Explicação mais provável	p. 44
3.3.3	Máximo a posteriori	p. 49
3.4	Redes semi-qualitativas	p. 49
4	Inferências usando programação multilinear	p. 56
4.1	Introdução	p. 56
4.2	Programação multilinear	p. 57
4.3	O algoritmo de reformulação-linearização	p. 58
4.4	Algoritmos para inferências em redes credais	p. 63
4.4.1	Transformando inferências em problemas multilineares	p. 64

4.5	Inferências em redes semi-qualitativas	p. 72
4.6	Soluções aproximadas para inferência em redes credais	p. 74
4.6.1	A/R+ para redes multi-conectadas	p. 75
4.6.2	A/R++	p. 76
4.6.3	Busca Local Iterativa	p. 78
4.7	Resultados experimentais	p. 81
4.7.1	Busca Local Iterativa	p. 81
4.7.2	RL e A/R++	p. 82
5	Considerações finais	p. 85
	Referências	p. 88

Lista de Figuras

1	Problema clássico do cão.	p. 16
2	Versão credal do problema do cão.	p. 19
3	Rede com topologia de largura limitada sem ciclos.	p. 24
4	Rede com topologia de largura limitada e ciclos.	p. 24
5	Rede com topologia de árvore.	p. 25
6	Rede com topologia de árvore e largura não limitada.	p. 25
7	Rede com largura não limitada.	p. 26
8	Árvore de seletores e variáveis booleanas, com variável extra para consulta.	p. 35
9	Árvore de seletores e variáveis booleanas.	p. 38
10	Rede utilizada no Teorema 3.42.	p. 51
11	Rede utilizada no Teorema 3.44.	p. 53
12	Rede com topologia em losango.	p. 70
13	Um exemplo de rede semi-qualitativa.	p. 73
14	Rede multi-conectada com seis vértices.	p. 76
15	Possível árvore de eliminação das variáveis para a rede da Figura 14.	p. 77
16	Uma rede credal com 13 nós e topologia em árvore.	p. 81

Lista de Tabelas

1	Resultados de complexidade em redes Bayesianas (BN).	p. 30
2	Resultados de complexidade em redes credais (CN).	p. 37
3	Resposta da inferência em uma SQPN de acordo com os resultados dos problemas SQPN-M e SQPN-m.	p. 50
4	Formato dos conjuntos de testes.	p. 82
5	Tamanho médio dos problemas multilineares e suas versões linearizadas correspondentes. Erros induzidos e média de nós avaliados.	p. 83
6	Testes com redes geradas aleatoriamente.	p. 84

1 *Preliminares*

1.1 Introdução e objetivo

Modelos probabilísticos têm se firmado na última década como ferramentas básicas em tomada de decisão e inteligência artificial. Por exemplo, o livro (RUSSELL; NORVIG, 2002) baseia toda a parte de manipulação de incertezas na teoria de probabilidades (RUSSELL *et al.*, 1995). Devido à necessidade de representar modelos com grande quantidade de variáveis aleatórias e número elevado de relações de independência, esquemas baseados em grafos têm sido desenvolvidos nos últimos anos, obtendo uma forma simplificada de modelar as relações entre as variáveis aleatórias. Esses esquemas permitem a representação e manipulação de grande número de variáveis e elevado número de relações de independência (CANO; CANO; MORAL, 1994; CANO; DELGADO; MORAL, 1993; COZMAN, 2000a; FAGIUOLI; ZAFFALON, 1998; GAAG, 1991; JENSEN, 1996; LUO *et al.*, 1996; SHAFER, 1996; TESSEM, 1992).

No entanto, com a utilização da teoria usual de probabilidades, não é considerada a possibilidade de incerteza nas próprias medidas de probabilidade. Na maioria dos casos não é simples atribuir valores precisos para a probabilidade de um evento. Neste sentido, uma das técnicas promissoras que surgiram nas últimas décadas é a representação de incerteza por conjuntos de probabilidades (WALLEY, 1996). Além de permitir a expressão de modelos probabilísticos usuais, a teoria de conjuntos de probabilidades permite que sejam construídos modelos para opiniões subjetivas baseadas em intervalos, tais como “a probabilidade de quebra do motor está entre 0.2 e 0.3”.

A introdução de conjuntos de probabilidades permite estudar robustez em modelos probabilísticos, investigar o comportamento/opiniões de grupos de especialistas, representar conhecimentos qualitativos, incompletos ou vagos (WALLEY, 1991). Uma rede credal é uma representação para este tipo de modelo, pois ela representa um conjunto de medi-

das de probabilidade através de um grafo acíclico orientado (que define as relações de independência entre as variáveis) e uma coleção de conjuntos locais de medidas de probabilidade (CANO; DELGADO; MORAL, 1993; COZMAN, 2000a; FAGIUOLI; ZAFFALON, 1998).

A teoria de conjuntos de probabilidades já adquiriu certa maturidade teórica, mas ainda não existem algoritmos para manipulação eficiente dos modelos baseados nessa teoria. Em geral os melhores algoritmos conhecidos ainda estão aquém de produzir soluções satisfatórias para problemas em escala prática.

O objetivo desse trabalho é estudar a complexidade dos problemas relacionados a redes credais e produzir algoritmos para inferência com conjuntos de probabilidades, focando principalmente em modelos baseados no conceito de independência forte. Nesse sentido, mostramos que técnicas de otimização não linear (principalmente programação multilinear) contribui na elaboração de novos algoritmos para inferência em problemas de porte prático. Até o momento a literatura não tem explorado esta possibilidade.

A análise de complexidade dos problemas e a investigação de técnicas de otimização matemática para inferência são as maiores contribuições deste trabalho. A análise de complexidade mostra que os problemas são difíceis, e que técnicas elaboradas precisam ser utilizadas (Capítulo 3). Os resultados obtidos utilizando idéias da otimização não linear são suficientes para afirmar que algoritmos de otimização global para programação não linear aplicados em inferências sobre conjuntos de probabilidades conseguem tratar problemas maiores que os demais algoritmos existentes (Capítulo 4). Além disso, a transformação da inferência em um problema não linear serve como base para a construção de algoritmos aproximados. A descrição destes procedimentos e resultados experimentais são apresentados.

1.2 Visão geral do documento

A teoria de redes credais, que pode ser vista como uma extensão das redes Bayesianas, possibilita uma maneira flexível para o tratamento de incertezas, pois é capaz de tratar crenças precisas, imprecisas, qualitativas e comparativas. Podemos ver as redes credais como uma ferramenta unificadora de diversos modelos gráficos para a representação de incerteza. No Capítulo 2 apresentamos os conceitos sobre as redes. Na Seção 2.2 trata-

mos das redes Bayesianas, na Seção 2.3 sobre as propriedades básicas das redes credais, incluindo o conceito de independência forte (Seção 2.3.3), e na Seção 2.4 as redes semi-qualitativas. Introduzimos na Seção 2.5 tipos de rede com relação à topologia e finalmente na Seção 2.6 apresentamos informalmente as inferências sobre as redes.

No Capítulo 3 apresentamos resultados de complexidade de diversos problemas sobre redes Bayesianas, credais e semi-qualitativas. Introduzimos ainda a definição e análise de complexidade de novos problemas ainda não explorados na literatura, como *maximin* a posteriori em redes Bayesianas, explicação mais provável e máximo a posteriori em redes credais. As seções 3.2, 3.3 e 3.4 tratam respectivamente dos problemas sobre redes Bayesianas, credais e semi-qualitativas.

Uma inferência marginal no contexto das redes credais é o cálculo de probabilidades máximas e mínimas para um dado evento. Os algoritmos existentes para inferências marginais são capazes de tratar apenas redes pequenas; os algoritmos criados e apresentados neste trabalho conseguem atingir resultados mais satisfatórios. No Capítulo 4 apresentamos um algoritmo exato para inferências, baseado em programação multilinear, e algumas soluções aproximadas. Na Seção 4.2 descrevemos o funcionamento do algoritmo exato, e nas seções 4.4 e 4.5 mostramos sua aplicação para redes credais e semi-qualitativas. Também apresentamos dois algoritmos para inferência aproximada, descritos na Seção 4.6. Apresentamos o algoritmo A/R++, uma extensão do algoritmo A/R+ de (ROCHA; COZMAN; CAMPOS, 2003). Essencialmente estendemos o algoritmo A/R+ em dois pontos: tratamos de redes multi-conectadas e aumentamos as informações transmitidas usando idéias da programação multilinear (Seção 4.6.2). Aproximações com este novo algoritmo são significativamente melhores que aquelas produzidas pelo algoritmo A/R+, além de sua aplicação ser mais abrangente. O outro algoritmo aproximado é o ILS (*Busca Local Iterativa*), descrito na Seção 4.6.3. Ele obtém resultados precisos de forma muito eficiente. Finalizamos o capítulo com a Seção 4.7, apresentando os resultados experimentais dos algoritmos desenvolvidos. Na Seção 4.7.1, citamos os resultados obtidos com o algoritmo ILS e, na Seção 4.7.2, apresentamos experimentos realizados com os algoritmos RL (para programação multilinear) e A/R++.

No Capítulo 5 concluímos o trabalho, descrevemos os resultados obtidos e discutimos caminhos futuros.

2 Modelos gráficos para inferência

2.1 Introdução

Este capítulo apresenta as redes tratadas neste trabalho e as definições de independência utilizadas em tais redes. Focamos os estudos sobre as redes Bayesianas, as redes credais e as redes semi-qualitativas. Estas redes são modelos gráficos para representação de incerteza através de variáveis aleatórias.

As redes Bayesianas usam a teoria de probabilidades para representar incertezas, apresentando explicitamente as dependências condicionais entre as diferentes variáveis. A introdução de imprecisão nas probabilidades através das redes credais permite estudar robustez em modelos probabilísticos, investigar o comportamento/opiniões de grupos de especialistas, representar conhecimentos incompletos ou vagos, entre outros tópicos. As redes semi-qualitativas introduzem a idéia de utilizar tanto restrições quantitativas (das redes Bayesianas) quanto restrições qualitativas sobre um mesmo modelo.

As Seções 2.2, 2.3 e 2.4 descrevem as redes Bayesianas, credais e semi-qualitativas, respectivamente. A Seção 2.3.3 apresenta os tipos de independência comumente adotadas para as redes credais e semi-qualitativas. Além da divisão das redes de acordo com suas probabilidades e restrições que permitem, introduzimos três tipos de redes de acordo com a topologia. Esses tipos são discutidos na Seção 2.5. Então a Seção 2.6 introduz brevemente os problemas de inferência que estamos interessados. Esses problemas são adequadamente descritos no Capítulo 3.

2.2 Redes Bayesianas

Uma rede Bayesiana (ou BN para simplificar) representa uma única densidade de probabilidade conjunta sobre uma coleção de variáveis aleatórias. Assumimos neste texto que as variáveis são categóricas, isto é, são discretas e seus possíveis estados são chamados de categorias da variável. Utilizamos no decorrer do texto letras maiúsculas para indicar variáveis e letras minúsculas para suas possíveis categorias.

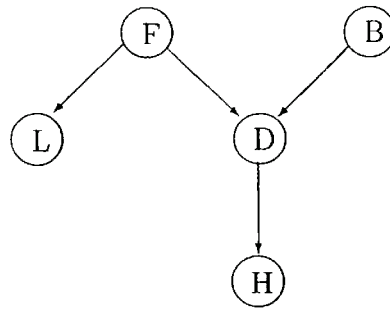
Definição 2.1 *Uma rede Bayesiana é uma tripla $(G, \mathbb{X}, \mathbb{P})$, onde $G = (V_G, E_G)$ é um grafo orientado acíclico, com V_G sua coleção de nós associados às variáveis aleatórias \mathbb{X} (um nó por variável), e E_G uma coleção de arcos definida sobre $V_G \times V_G$, relacionado os elementos de \mathbb{X} ; \mathbb{P} é uma coleção completa de densidades de probabilidade condicionais $p(X_i | \text{pa}(X_i))$ (números racionais), onde $\text{pa}(X_i)$ denota os pais de X_i no grafo ($\text{pa}(X_i)$ pode ser vazio), para todas as relações $\{X_i, \text{pa}(X_i)\}$ definidas por E_G .*

Assumimos, para todas as redes tratadas nesse trabalho, que o número de categorias de cada variável aleatória de \mathbb{X} é $O(|V_G|)$. Em uma rede Bayesiana toda variável é independente de seus não-descendentes não-pais dados seus pais (condição de Markov). Essa é uma condição que torna as redes Bayesianas interessantes, pois os cálculos de envolvendo as probabilidades são simplificados. Essa condição induz uma densidade de probabilidade conjunta pela expressão

$$p(X_1, \dots, X_n) = \prod_i p(X_i | \text{pa}(X_i)). \quad (2.1)$$

Denotamos por \mathbb{E} o conjunto de observações de uma rede Bayesiana (chamado de *evidência* da rede). Por exemplo, $\mathbb{E} = \{X_1 = x_1, X_4 = x_4\}$ fixa os valores das variáveis aleatórias X_1 e X_4 . O símbolo $\mathbb{X}_{\mathbb{E}}$ denota a coleção de variáveis que têm seus valores fixados pela evidência \mathbb{E} . No exemplo, temos $\mathbb{X}_{\mathbb{E}} = \{X_1, X_4\}$. Assumimos que todo evento condicionante tem probabilidade maior ou igual a zero. Deixamos para trabalhos futuros o caso onde os eventos condicionantes podem ter probabilidade zero (COLETTI; SCOZZAFAVA, 2002; COZMAN, 2002; VANTAGGI, 2003).

Uma rede é dita *binária* se todos os elementos de \mathbb{X} são variáveis aleatórias booleanas. Normalmente denotamos por x e \bar{x} as possíveis categorias de uma variável booleana X .



$$\begin{aligned}
 P(F = f) &= 0.4 & P(D = d|F = f, B = b) &= 0.8 \\
 P(B = b) &= 0.5 & P(D = d|F = f, B = \bar{b}) &= 0.1 \\
 P(L = l|F = f) &= 0.6 & P(D = d|F = \bar{f}, B = b) &= 0.1 \\
 P(L = l|F = \bar{f}) &= 0.05 & P(D = d|F = \bar{f}, B = \bar{b}) &= 0.7 \\
 & & P(H = h|D = d) &= 0.6 \\
 & & P(H = h|D = \bar{d}) &= 0.3
 \end{aligned}$$

Figura 1: Problema clássico do cão.

Na Figura 1 vemos uma rede Bayesiana binária modelando o problema clássico do cão (CHARNIAK, 1991). Este problema relaciona a probabilidade da família não estar em casa (variável booleana F ou *family-out*), a probabilidade da luz estar acesa (variável booleana L ou *light-on*), a probabilidade do cão estar fora da casa (variável booleana D ou *dog-out*), a probabilidade do cão estar com problema de intestino (variável booleana B ou *bowel-problem*), e a probabilidade de latidos (variável booleana H ou *hear-bark*). Note que cada variável $X \in \{B, F, L, D, H\}$ tem as categorias x e \bar{x} . Sempre que não houver dúvidas, utilizaremos uma notação simplificada, como por exemplo $P(\bar{l})$ no lugar de $P(L = \bar{l})$ e $P(\bar{l}|\bar{f})$ no lugar de $P(L = \bar{l}|F = \bar{f})$.

Podemos calcular a probabilidade conjunta desta rede de acordo com a Equação 2.1:

$$P(B, F, L, D, H) = P(H|D) P(D|F, B) P(L|F) P(F) P(B) \quad (2.2)$$

para valores fixados das variáveis B , F , L , D e H .

A construção de uma rede Bayesiana requer uma grande quantidade de definições precisas com relação às probabilidades dos eventos. Todas as probabilidades relacionadas à rede precisam ser fixadas. Diversas maneiras de relaxar essa imposição, que é muito forte em grande parte dos casos, já foram propostas, como a teoria de Dempster–Shafer (DEMPS-TER, 1968; SHAFER, 1976, 1990) e a teoria de possibilidades (GEBHARDT; KRUSE,

1995). Outra estratégia é associar intervalos de probabilidades com as variáveis da rede (KYBURG JR., 1987). Neste caso deparamos com alguns problemas, pois em diversos casos a regra de Bayes não pode ser diretamente aplicada (CHRISMAN, 1996). A teoria de redes credais (COZMAN, 2000a) e as redes semi-qualitativas (WELLMAN, 1990) são alternativas interessantes, pois probabilidades condicionais podem ser facilmente calculadas através da aplicação da regra de Bayes.

2.3 Conjuntos e redes credais

2.3.1 Conjuntos credais

Conjuntos de medidas de probabilidade oferecem uma linguagem geral que pode encapsular diversos modelos de interesse em inteligência artificial, tais como funções de crenças (*belief functions*), afirmações qualitativas sobre probabilidades, restrições de lógica probabilística (WALLEY, 1996).

Um conjunto fechado de distribuições de probabilidade é chamado de um conjunto credal (LEVI, 1980).

Definição 2.2 *Um conjunto fechado de densidades de probabilidade $p(X)$ para uma variável aleatória X é chamado de conjunto credal (credal set).*

Um conjunto credal para a variável X é denotado por $K(X)$.

Definição 2.3 *Dado um conjunto credal $K(X)$ e um evento A , a probabilidade superior (upper probability) e a probabilidade inferior (lower probability) de A são respectivamente os valores*

$$\begin{aligned}\bar{P}(A) &= \max_{p(X) \in K(X)} P(A), \\ \underline{P}(A) &= \min_{p(X) \in K(X)} P(A).\end{aligned}$$

Um conjunto credal condicional $K(X|A)$ é formado por uma coleção de distribuições condicionais $p(X|A)$, obtida aplicando-se a regra de Bayes para cada distribuição do conjunto de probabilidades conjuntas (WALLEY, 1991).

Assumimos que, dado um conjunto credal, encontrar a probabilidade superior ou inferior

sobre esse conjunto é uma operação polinomial sobre a definição (vértices ou restrições) do conjunto.

As mesmas idéias podem ser aplicadas para a definição de expectativa inferior e superior:

Definição 2.4 *Dado um conjunto credal $K(X)$ e uma função f sobre X , a expectativa superior (upper expectation) e a expectativa inferior (lower expectation) de f são respectivamente os valores*

$$\begin{aligned}\bar{E}[f(X)] &= \max_{\mathcal{P}(X) \in K(X)} E_p[f(X)], \\ \underline{E}[f(X)] &= \min_{\mathcal{P}(X) \in K(X)} E_p[f(X)].\end{aligned}$$

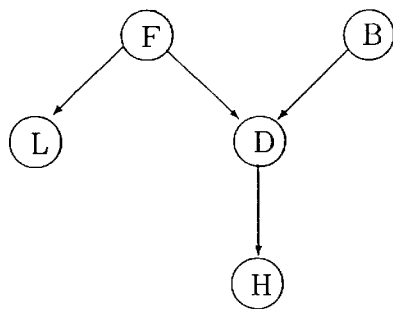
2.3.2 Redes credais

Uma rede credal generaliza o conceito de rede Bayesiana, permitindo que cada variável seja associada a um conjunto de densidades de probabilidade (no lugar de uma simples densidade, como nas redes Bayesianas) (CANO; DELGADO; MORAL, 1993; COZMAN, 2000a). Tais redes podem ser vistas como redes Bayesianas com parâmetros relaxados; podem ser utilizadas para estudar robustez de modelos probabilísticos, investigar o comportamento de grupos de especialistas, representar conhecimento vago ou incompleto sobre as probabilidades.

Uma rede credal é definida sobre um grafo orientado acíclico onde os nós estão associados a variáveis aleatórias e coleções de conjuntos credais.

Definição 2.5 *Uma rede credal é uma tripla $(G, \mathbb{X}, \mathbb{K})$, onde $G = (V_G, E_G)$ é um grafo orientado acíclico, com V_G sua coleção de nós associados às variáveis aleatórias \mathbb{X} (um nó por variável), e E_G uma coleção de arcos sobre $V_G \times V_G$, relacionando os elementos de \mathbb{X} ; \mathbb{K} é uma coleção de conjuntos credais condicionais $K(X_i | \text{pa}(X_i))$, onde $\text{pa}(X_i)$ denota os pais de X_i no grafo ($\text{pa}(X_i)$ pode ser vazio), para todas as relações $\{X_i, \text{pa}(X_i)\}$ definidas por E_G .*

Quando os conjuntos credais condicionais de \mathbb{K} são independentes de nó para nó da rede, dizemos que a rede é *localmente especificada*. Assumimos que todas as redes são localmente especificadas. A condição de Markov também é válida: toda variável é independente de seus não-descendentes não-pais dados seus pais.



$$\begin{array}{ll}
 0.4 \leq P(f) \leq 0.5 & 0.8 \leq P(d|f, b) \leq 0.8 \\
 0.5 \leq P(b) \leq 0.6 & 0.05 \leq P(d|f, \bar{b}) \leq 0.1 \\
 0.6 \leq P(l|f) \leq 0.6 & 0.05 \leq P(d|\bar{f}, b) \leq 0.2 \\
 0.05 \leq P(l|\bar{f}) \leq 0.05 & 0.7 \leq P(d|\bar{f}, \bar{b}) \leq 0.7 \\
 & 0.6 \leq P(h|d) \leq 0.6 \\
 & 0.3 \leq P(h|\bar{d}) \leq 0.3
 \end{array}$$

Figura 2: Versão credal do problema do cão.

Conjuntos credais $K(X|\text{pa}(X))$ sobre uma variável X da rede, para diferentes configurações de $\text{pa}(X)$, são ditos *separadamente especificados* quando não existem restrições no conjunto $K(X|\text{pa}(X) = \pi_1)$ baseadas nas propriedades de $K(X|\text{pa}(X) = \pi_2)$, para quaisquer instanciações $\pi_2 \neq \pi_1$ dos pais de X — isto é, temos um conjunto credal condicional para cada configuração dos pais e esses conjuntos não têm relação entre eles.

Na Figura 2 apresentamos uma rede similar àquela da Figura 1, mas com relaxações nas definições das probabilidades condicionais. Note que agora as probabilidades não são definidas por simples valores, mas por conjuntos. Por exemplo, $P(f)$ e $P(d|\bar{f}, b)$ são de fato desconhecidos; conhecemos apenas algumas restrições sobre os possíveis valores.

2.3.3 Conceitos de independência

Quando lidamos com redes Bayesianas, não há dúvida sobre a definição de independência, proveniente dos axiomas de probabilidade. Todavia, para as redes credais e semi-qualitativas, que modelam imprecisões sobre os valores de probabilidade, existem diferentes conceitos de independência, como a independência forte e a independência epistêmica (CAMPOS; MORAL, 1995; COUSO; MORAL; P., 2000). Ambos são equivalentes sobre as redes Bayesianas, mas diferem nas demais. Adotamos neste trabalho o conceito de

independência forte.

No conceito de independência forte, duas variáveis X e Y são independentes quando todo ponto extremo do conjunto credal das distribuições conjuntas $K(X, Y)$ satisfaz independência estocástica entre X e Y (isto é, $p(X|Y) = p(X)$ e $p(Y|X) = p(Y)$) (COZMAN, 2000a).

Independência forte é o conceito mais adotado para conjuntos credais devido a sua relação direta com independência estocástica. A extensão forte de uma rede credal é o maior conjunto que satisfaz essas condições descritas (COZMAN, 2000c). Assim, obtemos a extensão forte de uma rede credal através do fecho convexo de todas as distribuições conjuntas que satisfazem a propriedade de Markov: toda variável é *fortemente* independente de seus não-descendentes não-pais dado seus pais. Cada vértice de uma extensão forte fatora-se como na Expressão (2.1):

$$p(X_1, \dots, X_n) = \prod_i p(X_i | \text{pa}(X_i))$$

Sob esse conceito, podemos ver uma rede credal como uma coleção de redes Bayesianas, cada uma sob o mesmo grafo, já que em cada vértice da extensão temos exatamente uma rede Bayesiana.

2.4 Redes qualitativas e semi-qualitativas

Iniciamos essa seção explicando o que são as redes qualitativas, para então introduzir o conceito das redes semi-qualitativas.

2.4.1 Redes qualitativas

Uma rede qualitativa (QPN) consiste de um grafo orientado acíclico, um conjunto de variáveis aleatórias onde cada variável está associada a um nó do grafo, e uma coleção de restrições qualitativas nos valores de probabilidades sobre as variáveis. O grafo descreve uma condição de Markov: toda variável é independente de seus não-descendentes não-pais dados seus pais. Assumimos neste trabalho que as variáveis das redes qualitativas são booleanas, embora extensões simples dos resultados também valham para redes com

variáveis não booleanas. Dizemos que as variáveis booleanas das redes qualitativas podem assumir uma categoria “alta” ou uma categoria “baixa”.

Restrições em uma QPN derivam de *influências* e *sinergias* entre valores de probabilidades (WELLMAN, 1990). Uma *influência* entre dois nós expressa como os valores de um nó influenciam as probabilidades dos valores do outro nó.

Definição 2.6 *Uma influência positiva de um nó A em seu efeito B , denotada por $S^+(A, B)$, expressa que a observação de valores altos para A faz com que valores altos de B sejam mais comuns, independente de qualquer outra influência direta em B . Ou seja,*

$$P(b|a, x) \geq P(b|\bar{a}, x) \quad (2.3)$$

para qualquer instanciização completa x de $\text{pa}(B) \setminus \{A\}$.

Influências negativas, denotadas por $S^-(A, B)$, e *influências estáveis* (ou *influência zero*), denotadas por $S^0(A, B)$, são definidas analogamente, substituindo \geq por \leq e $=$ respectivamente. Uma influência é dita monotônica quando ela é negativa, positiva ou estável. Se a influência de A em B não é monotônica, dizemos que é ambígua, e denotamos por $S^?(A, B)$.

Sinergias representam interações entre influências. Uma *sinergia aditiva* entre três nós expressa como os valores de dois nós influenciam juntamente as probabilidades dos valores do terceiro nó.

Definição 2.7 *Uma sinergia aditiva positiva dos nós A e B em seu efeito comum C , denotada por $Y^+(\{A, B\}, C)$, expressa que a influência conjunta de A e B em C é maior que a soma de suas influências separadas, independentemente de outras influências em C . Ou seja,*

$$P(c|a, b, x) + P(c|\bar{a}, \bar{b}, x) \geq P(c|\bar{a}, b, x) + P(c|a, \bar{b}, x) \quad (2.4)$$

para toda instanciização completa x de $\text{pa}(C) \setminus \{A, B\}$.

Sinergias aditivas negativas, *sinergias aditivas zero* ou *estáveis*, e *sinergias aditivas ambíguas* são definidas analogamente.

Uma *sinergia produto* expressa como o valor de um nó influencia as probabilidades dos valores de outro nó, dado o valor de um terceiro.

Definição 2.8 *Uma sinergia produto positiva de um nó A em um nó B dado C , denotada por $X^+(\{A, B\}, C)$, significa que valores mais altos de A indicam que valores mais altos de B são mais freqüentes:*

$$P(c|a, b, x) P(c|\bar{a}, \bar{b}, x) \geq P(c|a, \bar{b}, E) P(c|\bar{a}, b, x) \quad (2.5)$$

para toda instanciação completa x de $\text{pa}(C) \setminus \{A, B\}$.

Sinergias produto negativas, sinergias produto zero, e sinergias produto ambíguas são definidas analogamente.

Existem diversas extensões para as QPNs (BOLT; RENOIJ; GAAG, 2003; RENOIJ; GAAG, 1999; RENOIJ; GAAG; PARSONS, 2000). Se a diferença $P(b|ax) - P(b|\bar{a}x)$ leva a sinais contraditórios dependendo da instanciação x de $\text{pa}(B) \setminus \{A\}$ (isto é, diferentes instanciações levam a diferentes sinais), a influência de A em B é não monotônica. Mas se olharmos para a diferença $P(b|ax) - P(b|\bar{a}x)$ separadamente para cada x , a influência de A em B é sempre não ambígua, ou seja, é positiva, negativa, ou zero. Para capturar o sinal de uma influência não monotônica para cada x , (BOLT; GAAG; RENOIJ, 2003) introduzem o conceito de sinais situacionais.

Definição 2.9 *A influência situacional positiva de A em B dada alguma evidência e sobre $\text{pa}(B) \setminus \{A\}$, denotada por $S_e^{?(+)}(A, B)$, indica que*

$$P(b|ae) \geq P(b|\bar{a}e). \quad (2.6)$$

Influências situacionais negativas e zero (ou estáveis) são definidas analogamente. Assim, para cada instanciação x de $\text{pa}(B) \setminus \{A\}$, podemos ter uma influência situacional diferente. Sinais situacionais representam uma maneira geral de tratar influências qualitativas, separando claramente influências ambíguas de influências não monotônicas. Em uma QPN padrão, uma influência não monotônica aparece como ambígua, já que não é possível fazer a distinção. Porém informações sobre as influências não monotônicas podem levar a resultados mais precisos.

Além disso, (RENOIJ; GAAG, 1999) introduz o formalismo “aumentado” para redes qualitativas, definindo a noção de influência fraca e forte.

Definição 2.10 *Uma influência fraca positiva de A em seu filho B indica que*

$$0 \leq P(b|a, x) - P(b|\bar{a}, x) \leq \delta \quad (2.7)$$

para toda instanciação completa x de $\text{pa}(B) \setminus \{A\}$, onde δ é um valor limitante de corte.

Definição 2.11 *Uma influência forte positiva de A em B afirma que*

$$P(b|a, x) - P(b|\bar{a}, x) \geq \delta \geq 0 \quad (2.8)$$

para toda instanciação completa x de $\text{pa}(B) \setminus \{A\}$.

Influências fracas negativas e influências fortes negativas são definidas analogamente.

Adotamos uma definição de QPN que engloba as *redes qualitativas situacionais* de (BOLT; GAAG; RENOOIJ, 2003) e as *redes qualitativas aumentadas* de (RENOOIJ; GAAG, 1999):

Definição 2.12 *Uma rede qualitativa (QPN) é uma tripla $(G, \mathbb{X}, \mathbb{R})$, onde $G = (V_G, E_G)$ é um grafo orientado acíclico, com V_G sua coleção de nós associados às variáveis aleatórias \mathbb{X} (um nó por variável), e E_G uma coleção de arcos definida sobre $V_G \times V_G$, relacionado os elementos de \mathbb{X} ; \mathbb{R} é uma coleção de relações qualitativas (influências qualitativas, sinergias aditivas, sinergias produto, influências situacionais, influências fracas e fortes) entre X_i e $\text{pa}(X_i)$ ($\text{pa}(X_i)$ pode ser vazio), para todos $\{X_i, \text{pa}(X_i)\}$ definidos por E_G .*

2.4.2 Redes semi-qualitativas

(PARSONS; DOHNAL, 1993) e (RENOOIJ; GAAG, 2002) propõe redes semi-qualitativas (SQPN) que misturam regras quantitativas e qualitativas. Utilizamos uma versão mais geral para essa definição, englobando todos os formalismos citados na Seção 2.4.1 para as relações qualitativas. Uma SQPN consiste em um grafo orientado acíclico com variáveis aleatórias associadas aos seus nós e uma condição de Markov, onde cada nó X está associado a distribuições condicionais $P(X|\text{pa}(X))$ ou a afirmações qualitativas das QPNs.

Definição 2.13 *Uma rede semi-qualitativa é uma tripla $(G, \mathbb{X}, \mathbb{A} = \{\mathbb{P}, \mathbb{R}\})$, onde $G = (V_G, E_G)$ é um grafo orientado acíclico com V_G sua coleção de nós associados às variáveis aleatórias \mathbb{X} (um nó por variável), e E_G uma coleção de arcos definida sobre $V_G \times V_G$, relacionado os elementos de \mathbb{X} ; \mathbb{A} é uma coleção de restrições sobre as variáveis da rede e seus pais, respeitando as relações definidas por E_G . Para cada variável X_i , ou estão*



Figura 3: Rede com topologia de largura limitada sem ciclos.

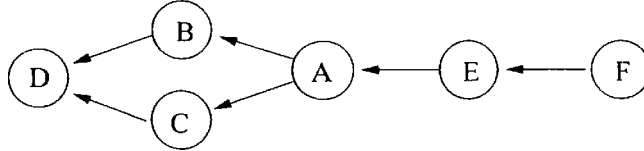


Figura 4: Rede com topologia de largura limitada e ciclos.

definidas as densidades de probabilidade condicionais $p(X_i|\text{pa}(X_i))$ em \mathbb{P} , ou estão especificadas relações qualitativas das QPNs entre X_i e $\text{pa}(X_i)$ em \mathbb{R} (ambas não acontecem ao mesmo tempo).

Esta é uma definição mais geral de SQPNs que outras encontradas na literatura (PARSONS; DOHNAL, 1993; RENOOLJ; GAAG, 2002). Assim SQPNs oferecem uma combinação de QPNs e redes Bayesianas. Adotamos também, para as SQPNs, o conceito de independência forte descrito na Seção 2.3.3.

2.5 Redes de acordo com a topologia

Consideramos neste trabalho três tipos de redes classificadas de acordo com suas topologias: a classe de redes que tem grau e largura limitada por uma função logarítmica no tamanho da entrada, a classe de redes que não contém nenhum tipo de ciclo (orientado ou não) e as demais.

Utilizamos a definição de *tree-width* para definir a largura de um grafo (KLOKS, 1994).

Definição 2.14 *Seja k um inteiro positivo. Uma k -árvore é um grafo de $n > k$ vértices definido recursivamente como segue. Um clique com $k + 1$ vértices é uma k -árvore. Dada uma k -árvore T com n vértices, uma k -árvore com $n + 1$ vértices pode ser construída adicionando-se um novo vértice em T , conectando-o a todos os k vértices de um k -clique de T .*

Definição 2.15 *A largura (ou induced-width) de um grafo G é o menor k para o qual G*

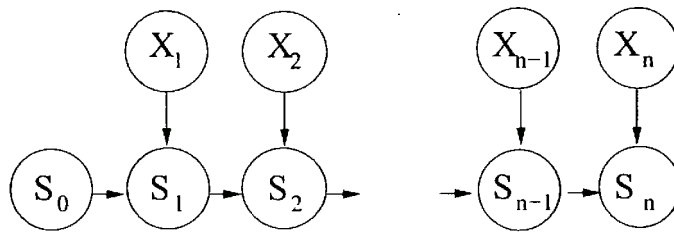


Figura 5: Rede com topologia de árvore.

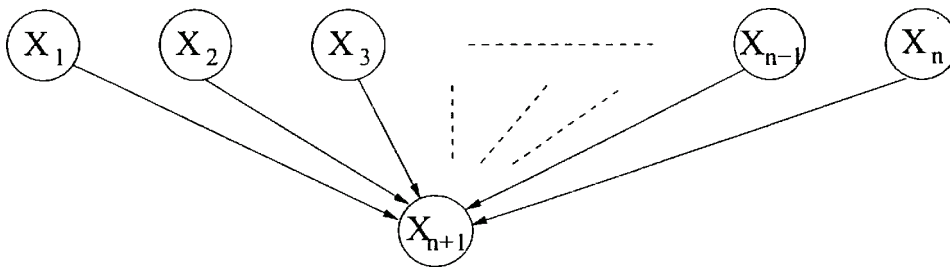


Figura 6: Rede com topologia de árvore e largura não limitada.

é subgrafo de uma k -árvore.

Definição 2.16 Seja $f(s)$ uma função polinomial sobre o tamanho s da entrada. Uma rede Bayesiana $(G, \mathbb{X}, \mathbb{P})$ tem largura limitada (também chamada de BIW ou bounded induced-width) quando G tem largura $\log(f(s))$ e o grafo subjacente de G (desprezando as orientações dos arcos) tem grau máximo $\log(f(s))$.

Note que esta definição não exige que a largura da rede seja constante; permitimos que a largura varie com o tamanho da rede. As Figuras 3 e 4 mostram exemplos simples de redes com largura limitada. Analisamos também uma subclasse das redes de largura limitada:

Definição 2.17 Uma rede Bayesiana $(G, \mathbb{X}, \mathbb{P})$ é dita uma árvore se o grafo subjacente a G (desprezando as orientações dos arcos) não contém ciclos.

As Figuras 5 e 6 têm exemplos de redes com topologia de árvore. Nos demais casos dizemos que a rede é *multi-conectada*. A Figura 6 trás um exemplo onde claramente a largura da rede não é limitada, apesar de ser uma árvore. A Figura 7 tem um exemplo de rede multi-conectada. As redes Bayesianas, as redes credais e as redes semi-qualitativas podem ser classificadas como árvores, largura limitada e multi-conectadas, dependendo de suas topologias.

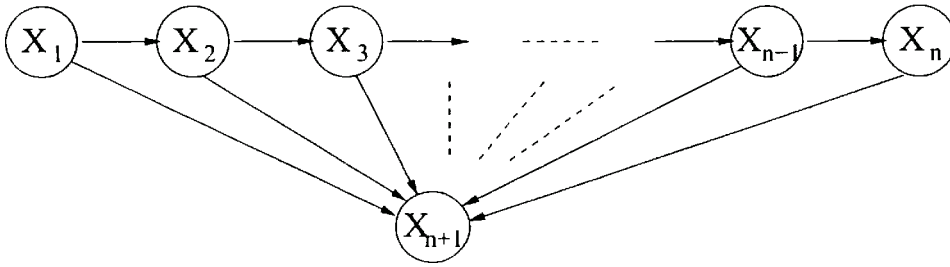


Figura 7: Rede com largura não limitada.

2.6 Inferência na presença de imprecisão

Nesta seção descrevemos informalmente a dificuldade de executar inferências na presença de imprecisão. Para tal, fazemos uma comparação entre uma inferência sobre uma rede Bayesiana e sua correspondente em uma rede credal.

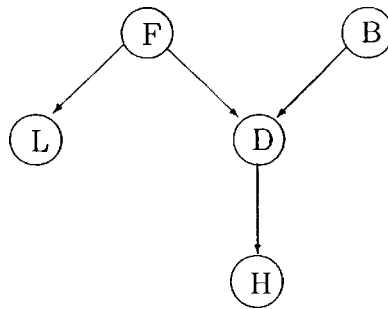
Inferências em redes Bayesianas normalmente envolvem o cálculo de probabilidades marginais para uma variável de consulta. Voltando ao problema do cão, apresentado na Figura 1 e redesenhado aqui por conveniência, podemos calcular a probabilidade da luz estar apagada $P(\bar{l})$:

$$\begin{aligned} P(\bar{l}) &= P(\bar{l}|f) P(f) + P(\bar{l}|\bar{f}) P(\bar{f}) \\ &= 0.4 \times 0.4 + 0.95 \times 0.6 \\ &= 0.73 \end{aligned}$$

Em uma rede credal, uma típica inferência sobre L seria o cálculo do menor valor que a probabilidade $P(\bar{l})$ pode assumir. O cálculo desse valor já apresenta dificuldades, pois precisamos descobrir quais probabilidades minimizam a probabilidade da luz estar apagada.

$$\underline{P}(\bar{l}) = \min P(\bar{l}) = \min [P(\bar{l}|f) P(f) + P(\bar{l}|\bar{f}) P(\bar{f})]$$

Neste pequeno exemplo (Figura 2), podemos visualmente notar que $P(\bar{l}|f) = 0.4$, $P(f) = 0.5$, $P(\bar{l}|\bar{f}) = 0.95$ e $P(\bar{f}) = 0.5$ minimizam a probabilidade desejada. Assim obtemos $\underline{P}(\bar{l}) = 0.675$. Isso nos diz que a probabilidade da luz estar apagada é pelo menos 67.5% (fazendo os cálculos podemos encontrar também a probabilidade máxima, que neste caso é



$$\begin{aligned}
 P(f) &= 0.4 \\
 P(b) &= 0.5 \\
 P(l|f) &= 0.6 \\
 P(l|\bar{f}) &= 0.05
 \end{aligned}$$

$$\begin{aligned}
 P(d|f, b) &= 0.8 \\
 P(d|f, \bar{b}) &= 0.1 \\
 P(d|\bar{f}, b) &= 0.1 \\
 P(d|\bar{f}, \bar{b}) &= 0.7 \\
 P(h|d) &= 0.6 \\
 P(h|\bar{d}) &= 0.3
 \end{aligned}$$

Versão Bayesiana

$$\begin{aligned}
 0.4 &\leq P(f) \leq 0.5 \\
 0.5 &\leq P(b) \leq 0.6 \\
 0.6 &\leq P(l|f) \leq 0.6 \\
 0.05 &\leq P(l|\bar{f}) \leq 0.05
 \end{aligned}$$

$$\begin{aligned}
 0.8 &\leq P(d|f, b) \leq 0.8 \\
 0.05 &\leq P(d|f, \bar{b}) \leq 0.1 \\
 0.05 &\leq P(d|\bar{f}, b) \leq 0.2 \\
 0.7 &\leq P(d|\bar{f}, \bar{b}) \leq 0.7 \\
 0.6 &\leq P(h|d) \leq 0.6 \\
 0.3 &\leq P(h|\bar{d}) \leq 0.3
 \end{aligned}$$

Versão credal

$\bar{P}(l) = 73\%$). Para realizar esses cálculos, estamos utilizando o conceito de independência forte descrito anteriormente.

Note que deixamos um problema onde a maior dificuldade era encontrar a melhor ordem para fazer os cálculos e chegamos em um problema de otimização. Essa mesma dificuldade existe para as redes semi-qualitativas. No Capítulo 4 tratamos em detalhes alguns algoritmos baseados em programação multilinear para resolver esses problemas de inferência.

3 *Resultados de complexidade*

3.1 Introdução

Este capítulo apresenta formalmente os problemas de inferências sobre os modelos de incerteza que iremos tratar neste trabalho. Apresentamos as definições de suas versões de decisão, além da análise de complexidade computacional.

Focamos os estudos sobre as redes Bayesianas, as redes credais e as redes semi-qualitativas. Atualmente já existe uma compreensão clara sobre a complexidade computacional de inferências em redes Bayesianas. Adicionamos a este contexto as redes BIW, uma nova classe que fortalece os resultados existentes. Além disso, apresentamos novos problemas, como o maximin a posteriori (MmAP), que pode ser de interesse em teoria de jogos.

Poucos resultados são conhecidos sobre inferências em redes credais. Apresentamos a complexidade da computação de probabilidades superiores e inferiores, explicação mais provável (MPE) e máximo a posteriori (MAP).

Mostramos que, surpreendentemente, o problema MPE sem observações em redes credais de largura limitada é polinomial, enquanto que o problema MPE com observações e o problema de MAP são Σ_2^P -completos para redes credais.

Aumentamos também a relação entre redes Bayesianas e redes credais à medida que apresentamos conexões entre a computação de limites de probabilidades nas redes credais e problemas de MAP em redes Bayesianas.

Esses resultados sugerem que a inclusão de intervalos/conjuntos leva-nos um passo adiante em termos de dificuldade computacional.

Apresentamos também consultas sobre redes semi-qualitativas. Além da inferência tradi-

Problema	<i>Árvore</i>	<i>Largura limitada</i>	<i>Multi-conectada</i>
Pr	Polinomial	Polinomial	PP-Completo
MPE	Polinomial	Polinomial	NP-Completo
MPEe	Polinomial	Polinomial	PP-Completo
MAP	NP-Completo	NP-Completo	NP ^{PP} -Completo
MmAP	Σ_2^P -Completo	Σ_2^P -Completo	NP ^{PP} -Difícil

Tabela 1: Resultados de complexidade em redes Bayesianas (BN).

cional, tratamos novos problemas como MPE e MAP. Mostramos ainda que os problemas em redes semi-qualitativas podem ser tratados pelas mesmas idéias utilizadas para as redes credais.

A Seção 3.2 trata dos problemas sobre redes Bayesianas. A Seção 3.3 contém os resultados para as redes credais, e a Seção 3.4 para as redes semi-qualitativas. As tabelas 1 e 2 resumem os resultados de complexidade mais relevantes.

Alguns comentários geral são importantes antes de iniciarmos: chamamos de tempo polinomial quando o problema pode ser resolvido em tempo polinomial no tamanho da entrada. Utilizamos simplesmente o termo *redução* para uma redução de tempo polinomial. Dizemos que um problema é *solucionado por* outro quando existe uma redução do primeiro para o segundo problema. Se essa última relação é simétrica, então dizemos que os problemas são *equivalentes*.

3.2 Redes Bayesianas

Nesta seção apresentamos as versões de decisão dos problemas de inferência em redes Bayesianas e suas complexidades.

3.2.1 Atualização de crenças

Atualização de crenças (ou *belief updating*) é a tarefa de encontrar a probabilidade marginal $P(q|\mathbb{E})$, onde \mathbb{E} é a evidência e $Q \in \mathbb{X}$ é uma variável da rede Bayesianas.

Definição 3.1 *Dada uma rede Bayesianas $(G, \mathbb{X}, \mathbb{P})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, uma variável de consulta $Q \in \mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$ e sua categoria q , e um número racional r , BN-Pr é a*

questão de decidir se $P(q|\mathbb{E}) > r$.

Através de uma redução do problema de maioria de satisfabilidade MAJSAT, (LITTMAN; GOLDSMITH; MUNDHENK, 1998) mostra resultados que indicam a validade do Teorema 3.2. O Teorema 3.9 apresentado na próxima seção também pode ser facilmente estendido para obter esse mesmo resultado. A complexidade da versão funcional é discutida em (ROTH, 1996).

Teorema 3.2 *BN-Pr é PP-Completo.*

O problema BN-Pr com consulta sobre várias variáveis também é PP-Completo, assim como se trocássemos a decisão para $P(q|\mathbb{E}) < r$, pois a classe PP é fechada sobre reduções de tempo polinomial sobre tabelas-verdade (FORTNOW; REINGOLD, 1991).

(DECHTER, 1996) descreve algoritmos com complexidade de tempo exponencial na largura da árvore de eliminação das variáveis (que depende da ordem que as variáveis são eliminadas). (EYAL, 2001) mostra como obter uma aproximação constante para a ordem ótima em tempo polinomial nas redes PT e BIW. Juntando esses dois resultados, temos o Corolário 3.3.

Corolário 3.3 *PT-BN-Pr e BIW-BN-Pr estão em P.*

Note que este resultado é ligeiramente diferente daquele apresentado em (DECHTER, 1996), pois estamos lidando com a largura da rede, no lugar da largura da árvore de eliminação (que depende da ordem escolhida). Note ainda que utilizamos abreviações para referenciar os problemas. Por exemplo PT-BN-Pr é o problema de *belief updating* em uma rede Bayesiana com topologia de árvore.

3.2.2 Explicação mais provável

A explicação mais provável (MPE ou *most probable explanation*) é a questão de encontrar uma instanciamento $x = (x_1, x_2, \dots, x_n)$ para todas as variáveis da rede tal que $p(x|\mathbb{E})$ seja máximo. Note que fazemos distinção entre o problema MPE com e sem evidência.

Definição 3.4 *Dada uma rede Bayesiana $(G, \mathbb{X}, \mathbb{P})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, e um número racional r , BN-MPE é a questão de decidir se existe uma instanciamento x para $\mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$ tal que $P(x, \mathbb{E}) > r$.*

Definição 3.5 Dada uma rede Bayesiana $(G, \mathbb{X}, \mathbb{P})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, e um número racional r , BN-MPEe é a questão de decidir se existe uma instânciação x para $\mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$ tal que $P(x|\mathbb{E}) > r$.

Pelos mesmos argumentos apresentados para o problema de atualização de crenças, os resultados de (DECHTER, 1996) e (EYAL, 2001) implicam no Corolário 3.6.

Corolário 3.6 PT-BN-MPE e BIW-BN-MPE estão em P.

Corolário 3.7 PT-BN-MPEe e BIW-BN-MPEe estão em P.

Prova Segue dos Corolários 3.3 e 3.6. \square

Por uma redução a partir do problema de cobertura de vértices VC, (SHIMONY, 1994) mostra o Teorema 3.8. Por outro lado, o Teorema 3.9 mostra que BN-MPEe é PP-Completo.

Teorema 3.8 BN-MPE é NP-Completo.

Teorema 3.9 BN-MPEe é PP-Completo.

Prova Pertinência em PP é obtida pois podemos reescrever $P(x|\mathbb{E}) > r$ como $\frac{P(x, \mathbb{E})}{P(\mathbb{E})} > r$, e temo que PP é fechado sobre reduções de tempo polinomial sobre tabelas-verdade.

Para mostrar que BN-MPEe é PP-difícil, reduzimos o problema de decisão #3SAT ($\geq 2^{n/2}$) para ele. Note que este problema é PP-Completo (BAILEY; DALMAU; KOLAITIS, 2001) e pode ser enunciado como segue:

Definição 3.10 Dado um conjunto de variáveis booleanas $X = \{X_1, \dots, X_n\}$ e uma fórmula 3CNF¹ $\phi(X)$ com cláusulas $\{C_1, \dots, C_m\}$, o problema #3SAT ($\geq 2^{n/2}$) é a questão de decidir se $\phi(X)$ é satisfeita por pelo menos $2^{n/2}$ de todas as possíveis instanciações de X .

Construímos uma rede Bayesiana com nós binários X_1, \dots, X_n (x_i e \bar{x}_i são suas categorias) e C_1, \dots, C_m (c_i e \bar{c}_i são as categorias), onde X_i não tem pais e tem distribuição uniforme a priori e C_i tem três pais (que são as variáveis contidas na cláusula C_i) com probabilidades respeitando a tabela verdade da cláusula.

Além disso, inserimos um nó binário extra Y aparecendo não-negado em todas as cláusulas. Existem 2^n instanciações com $\{Y = y\}$ satisfazendo ϕ . Isso garante que a

¹Forma normal conjuntiva com três variáveis por cláusula.

fórmula é satisfeita por pelo menos uma instanciação, e assim o denominador da Equação (3.1) nunca é zero.

Agora resolvemos o problema BN-MPEe com consultas X_1, \dots, X_n, Y e evidência $\{C_i = c_i\}$ para todo $1 \leq i \leq m$ (isso indica que todas as cláusulas C_i são verdadeiras). Então $P(X, Y | c_1, \dots, c_m)$ é igual a

$$\begin{aligned}
&= \frac{\frac{1}{2^{n+1}} P(c_1, \dots, c_m | X_1, \dots, X_n, Y)}{\sum_{X'_1, \dots, X'_n, Y'} P(c_1, \dots, c_m | X'_1, \dots, X'_n, Y') P(X'_1, \dots, X'_n, Y')} \quad (3.1) \\
&= \frac{\frac{1}{2^{n+1}} P(c_1, \dots, c_m | X_1, \dots, X_n, Y)}{\sum_{X'_1, \dots, X'_n, Y'} [P(c_1, \dots, c_m | X'_1, \dots, X'_n, Y') \frac{1}{2^{n+1}}]} \\
&= \frac{1}{\#sats} \text{ se } X_1, \dots, X_n, Y \text{ satisfaz } \phi \text{ e } 0 \text{ caso contrário.}
\end{aligned}$$

onde $\#sats$ é o número total de instanciações satisfazendo ϕ . Dessa forma, $\max P(X, Y | c_1, \dots, c_m) \leq \frac{1}{2^{n+2^{n/2}}}$ implica que a fórmula $\phi(X)$ é satisfeita por pelo menos $2^{n/2}$ de todas as instanciações de X . Note que se $\max P(X, Y | c_1, \dots, c_m) = \alpha$ e $\alpha > \frac{1}{2^{n+2^{n/2}}}$, então $P(X, Y | c_1, \dots, c_m) = \alpha$ para todas as instanciações que satisfazem ϕ , implicando que existem menos que $2^{n/2}$ instanciações de X satisfazendo $\phi(X)$. \square

3.2.3 Máximo a posteriori

O problema de máximo a posteriori (**MAP** ou *maximum a posteriori hypothesis*) é a tarefa de encontrar uma instanciação incompleta que maximiza sua probabilidade. Isto é, dado um subconjunto de variáveis da rede, queremos encontrar uma instanciação para esse subconjunto de variáveis tal que sua probabilidade seja máxima. As variáveis deste subconjunto são chamadas de variáveis de MAP.

Definição 3.11 Dada uma rede Bayesiana $(G, \mathbb{X}, \mathbb{P})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, um conjunto $Q \subseteq \mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$, e um número racional r , BN-MAP é a questão de decidir se existe uma instanciação q para Q tal que $P(q | \mathbb{E}) > r$.

(PARK, 2002) reduz o problema de máxima satisfabilidade MAXSAT para obter o Teorema 3.12. O Corolário 3.13 estende o resultado para redes BIW.

Teorema 3.12 PT-BN-MAP é NP-Completo.

Corolário 3.13 BIW-BN-MAP é NP-Completo.

Prova A dificuldade segue do fato que BIW-BN-MAP é uma generalização de PT-BN-MAP, que é NP-Completo (Teorema 3.12). Pertinência também é trivial, já que escolhidas as categorias das variáveis de MAP, temos que resolver um problema BIW-BN-Pr para verificar a resposta. BIW-BN-Pr é polinomial (Corolário 3.3). \square

O artigo (PARK; DARWICHE, 2004) mostra, a partir de uma redução do problema da existência da maioria de satisfabilidade E-MAJSAT, o resultado a seguir.

Teorema 3.14 BN-MAP é NP^{PP} -Completo.

3.2.4 Maximin a posteriori

Introduzimos agora o conceito de *maximin a posteriori*, que pode ser de interesse em aplicações envolvendo teoria de jogos com maximizadores e minimizadores (KAKADE; ORTIZ, 2004). Note que:

- o conceito de *minimax* tem implicações equivalentes, ficando no segundo nível da hierarquia polinomial.
- não tratamos dos casos *maximax* e *minimin*, pois estes problemas são semelhantes a BN-MAP.

Definição 3.15 Dada uma rede Bayesiana $(G, \mathbb{X}, \mathbb{P})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, os conjuntos $A \subseteq \mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$ e $B \subseteq \mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$, com $A \cap B = \emptyset$, um número racional r , BN-MmAP é a questão de decidir se existe uma instanciação a para A tal que $\min_b P(a, b | \mathbb{E}) > r$.

Corolário 3.16 BN-MmAP é NP^{PP} -difícil.

Prova Segue diretamente do fato de BN-MmAP ser uma generalização de BN-MAP, que é NP^{PP} -Completo (Teorema 3.14). \square

Teorema 3.17 PT-BN-MmAP e BIW-BN-MmAP são Σ_2^P -Completos.

Prova Pertinência de BIW-BN-MmAP (que garante a pertinência de PT-BN-MmAP) é trivial. Dada uma instanciação para as variáveis de MAP, precisamos resolver uma minimização sobre as variáveis B , que é um problema coNP-Completo (veja ele como um BIW-CN-Pr, usando o Lema 3.29 e Corolário 3.23). A dificuldade de PT-BN-MmAP (que garante a dificuldade de BIW-BN-MmAP) é obtida por uma redução de uma versão do problema QSAT₂ que é Σ_2^P -Completo (PAPADIMITRIOU, 1994):

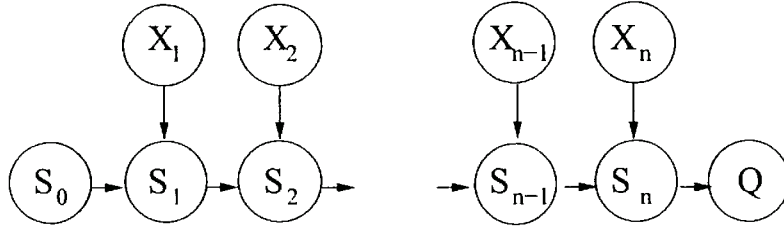


Figura 8: Árvore de seletores e variáveis booleanas, com variável extra para consulta.

Definição 3.18 *Dado um conjunto de variáveis $\mathbb{X} = \{X_1, \dots, X_n\}$, um inteiro $0 < k \leq n$ e uma fórmula booleana 3DNF² $\phi(\mathbb{X})$ sobre estas variáveis, $\text{QSAT}_2^{\text{DNF}}$ consiste em decidir se existe uma instanciação para as primeiras k variáveis tal que, para toda instanciação das demais $n - k$ variáveis, $\phi(\mathbb{X})$ é satisfeita.*

Construímos uma rede com topologia apresentada na Figura 8. Esta rede tem um nó binário para cada X_i , sem pais e com probabilidade a priori uniforme. Existem n nós S_i com pais S_{i-1} e X_i . Eles têm $m + 1$ categorias e suas probabilidades são definidas como segue (para $c \in \{1, \dots, m\}$ e $i \in \{1, \dots, n\}$):

$$\begin{aligned}
 P(S_i = c | S_{i-1} = c, x_i) &= \begin{cases} 1 & \text{se } x_i \in \text{cláusula } c \\ 0 & \text{se } \bar{x}_i \in \text{cláusula } c \\ \frac{1}{2} & \text{caso contrário} \end{cases} \\
 P(S_i = c | S_{i-1} \neq c, x_i) &= 0 \\
 P(S_i = c | S_{i-1} = c, \bar{x}_i) &= \begin{cases} 0 & \text{se } x_i \in \text{cláusula } c \\ 1 & \text{se } \bar{x}_i \in \text{cláusula } c \\ \frac{1}{2} & \text{caso contrário} \end{cases} \\
 P(S_i = c | S_{i-1} \neq c, \bar{x}_i) &= 0.
 \end{aligned}$$

Quando $S_i = 0$, temos as regras:

$$\begin{aligned}
 P(S_i = 0 | S_{i-1} = c, x_i) &= 1 - P(S_i = c | S_{i-1} = c, x_i) \\
 P(S_i = 0 | S_{i-1} = c, \bar{x}_i) &= 1 - P(S_i = c | S_{i-1} = c, \bar{x}_i) \\
 P(S_i = 0 | S_{i-1} = 0, X_i) &= 1 \text{ for } X_i \in \{x_i, \bar{x}_i\}.
 \end{aligned}$$

Essas regras garantem coerência nas probabilidades. Note que incluímos um nó extra S_0 com $P(S_0 = c) = \frac{1}{m+1}$ para todo c . Definimos também um nó binário Q com S_n sendo seu único pai, onde $P(Q | S_n = c) = 1$ se $c \neq 0$ e 0 caso contrário.

²Forma normal disjuntiva com três variáveis por cláusula.

Considere o cálculo de $P(S_n = c)$ para $c \neq 0$. Perceba que

$$P(S_n = c) = \sum_{X_n \in \{x_n, \bar{x}_n\}} P(S_n = c | S_{n-1} = c, X_n) P(X_n) P(S_{n-1} = c)$$

pois, para $c \neq 0$, toda vez que $\{S_i = c\}$ e $\{S_{i-1} \neq c\}$ aparecem juntos na fórmula o termo é levado a zero. Além disso, note que $P(X_i) = \frac{1}{2}$, independentemente dos outros valores.

$$P(S_n = c) = \frac{1}{2} \sum_{X_n \in \{x_n, \bar{x}_n\}} P(S_n = c | S_{n-1} = c, X_n) P(S_{n-1} = c)$$

Alterando a ordem dos cálculos, obtemos:

$$\begin{aligned} P(S_n = c) &= \frac{1}{2} P(S_{n-1} = c) A_{c,n} \\ &= \frac{1}{2^2} \left[\sum_{X_{n-1}} P(S_{n-1} = c | S_{n-2} = c, X_{n-1}) P(X_{n-1}) P(S_{n-2} = c) \right] A_{c,n} \\ &= \frac{1}{2^3} P(S_{n-2} = c) \left[\sum_{X_{n-1}} P(S_{n-1} = c | S_{n-2} = c, X_{n-1}) P(X_{n-1}) \right] A_{c,n} \\ &= \frac{1}{2^3} P(S_{n-2} = c) A_{c,n-1} A_{c,n} \\ &\dots \\ &= \frac{1}{2^n} P(S_0 = c) \prod_{i \in \{1, \dots, n\}} A_{c,i} \\ &= \frac{1}{2^n} \frac{1}{m+1} \prod_{i \in \{1, \dots, n\}} A_{c,i}, \end{aligned}$$

onde $A_{c,i} = \sum_{X_i} P(S_i = c | S_{i-1} = c, X_i)$ para $i \in \{1, \dots, n\}$. Note que, como estamos fazendo escolhas sobre X_i , $A_{c,i}$ vale de fato $P(S_i = c | S_{i-1} = c, X_i)$, de acordo com o X_i escolhido.

Dessa forma, temos que

$$P(X|q) = \frac{\frac{1}{2^n} P(q|X)}{P(q)} = \frac{\frac{1}{(m+1)2^n} \sum_{c \neq 0} \prod_i A_{c,i}}{\frac{m}{(m+1)} \left(\frac{1}{2}\right)^n} = \frac{\sum_{c \neq 0} \prod_i A_{c,i}}{m}.$$

$A_{c,i}$ assume valor zero apenas quando a variável que representa nega a cláusula c . Assim o numerador de $P(X|q)$ soma quantas cláusulas são satisfeitas pela instanciação das variáveis X .

Seja $X^- = \{X_1, \dots, X_k\}$ e $X^+ = \{X_{k+1}, \dots, X_n\}$. Dada uma instanciação das variáveis

Problema	<i>Árvore</i>	<i>Largura limitada</i>	<i>Multi-conectada</i>
Pr	NP-Completo	NP-Completo	NP ^{PP} -Completo
MPE	Polinomial	Polinomial	NP-Completo
MPEe	Σ_2^P -Completo	Σ_2^P -Completo	Σ_2^P -Difícil e PP-Difícil
MAP	Σ_2^P -Completo	Σ_2^P -Completo	NP ^{PP} -Difícil

Tabela 2: Resultados de complexidade em redes credais (CN).

X^- , temos que $\min_{X^+} P(X^-, X^+|q) = 0$ se existe uma instanciação para as variáveis X^+ que pode negar $\phi(X)$ (lembre que a fórmula está na DNF).

Dessa forma, questionar se o problema PT-BN-MmAP com variáveis de MAP X^- e evidência q tem resposta não zero é suficiente para resolver o problema QSAT₂^{DNF} apresentado. Neste caso a instanciação de X^- encontrada garante que todas as instanciações de X^+ irão satisfazer $\phi(X)$, pois $\min_{X^+} P(X|q)$ será maior que zero (independente da instanciação de X^+ , a fórmula será sempre satisfeita). \square

3.3 Redes credais

Definimos aqui os problemas de decisão para redes credais, incluindo problemas correspondentes àqueles apresentados na Seção 3.2 sobre redes Bayesianas.

3.3.1 Atualização de crenças

A atualização de crenças (CN-Pr ou *belief updating*) em redes credais é a inferência mais comumente encontrada na literatura.

Definição 3.19 *Dada uma rede credal $(G, \mathbb{X}, \mathbb{K})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, uma variável de consulta $Q \in \mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$ e sua categoria q , e um número racional r , CN-Pr é a questão de decidir se $\bar{P}(q|\mathbb{E}) > r$.*

Usamos consultas superiores na Definição 3.19. Todavia consultas inferiores também são de interesse. Podemos mostrar que ambas têm complexidades computacionais equivalentes:

Lema 3.20 *O cálculo de probabilidades marginais inferiores em uma rede credal é tão*

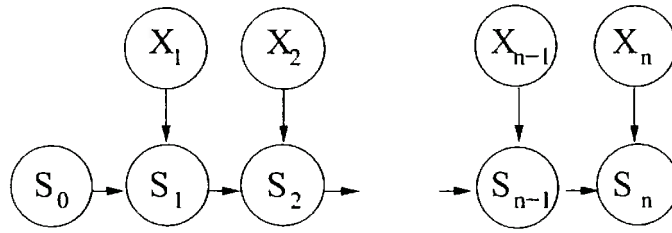


Figura 9: Árvore de seletores e variáveis booleanas.

difícil quanto o cálculo de probabilidades marginais superiores.

Prova Suponha que temos um problema CN-Pr com consulta $Q = q$. O cálculo de $\underline{P}(q|\mathbb{E})$ pode ser feito inserindo na rede um filho binário Q' em Q , onde $P(q'|Q) = 1$ se $Q \neq q$ e 0 caso contrário. Agora $\overline{P}(q'|\mathbb{E}) = \max \sum_{Q \neq q} P(Q|\mathbb{E}) = 1 - \underline{P}(q|\mathbb{E})$. Se desejarmos consultas superiores a partir de inferiores, a idéia é idêntica. \square

O problema PT-CN-Pr foi apresentado em (ROCHA; COZMAN, 2002) como sendo NP-Completo, mas a prova continha imprecisões. Além disso, o argumento central utilizava probabilidades zero, que em diversos casos devem ser evitadas. O Teorema 3.21 corrige essas dificuldades. As idéias dessa prova são utilizadas futuramente nos demais teoremas.

Teorema 3.21 PT-CN-Pr é NP-Completo.

Prova A pertinência de PT-CN-Pr é imediata, pois se forem dados os vértices de cada conjunto credal obtemos um problema PT-BN-Pr, que é resolvido em tempo polinomial.

Para mostrar que eles são NP-difíceis, reduzimos MAX-3-SAT para PT-CN-Pr. Utilizamos a seguinte formulação para o problema:

Definição 3.22 Dado um conjunto de variáveis booleanas $\{X_1, \dots, X_n\}$, uma fórmula 3CNF com cláusulas $\{C_1, \dots, C_m\}$ e um inteiro $0 \leq k < m$, o problema MAX-3-SAT é a questão de decidir se existe uma instância para as variáveis que satisfaz mais que k cláusulas da fórmula.

Inicialmente removemos todas as cláusulas que tem, ao mesmo tempo, x_i e \overline{x}_i , e decrementamos k para cada eliminação (afinal estas cláusulas já estão satisfeitas).

Para cada variável X_i , construímos dois nós, ditos X_i e S_i . O primeiro é binário, não tem pais e representa o estado da variável booleana X_i ; as probabilidades $P(X_i = x_i)$ e $P(X_i = \overline{x}_i)$ estão no intervalo $[\varepsilon, 1 - \varepsilon]$ ($0 < \varepsilon < \frac{1}{m+1}$ é uma constante pequena). O último

assume valores de 0 até m , ou seja, possui $m + 1$ categorias, tem S_{i-1} e X_i como pais e tem probabilidades definidas por

$$\begin{aligned} P(S_i = c | S_{i-1} = c, x_i) &= 0 \text{ se } x_i \in C_c, \text{ ou 1 caso contrário} \\ P(S_i = c | S_{i-1} = c, \bar{x}_i) &= 0 \text{ se } \bar{x}_i \in C_c, \text{ ou 1 caso contrário} \\ P(S_i = c | S_{i-1} \neq c, X_i) &= 0 \text{ para } X_i \in \{x_i, \bar{x}_i\}, \end{aligned}$$

para $c \neq 0$. Quando $S_i = 0$, temos as regras:

$$\begin{aligned} P(S_i = 0 | S_{i-1} = c, x_i) &= 1 - P(S_i = c | S_{i-1} = c, x_i) \\ P(S_i = 0 | S_{i-1} = c, \bar{x}_i) &= 1 - P(S_i = c | S_{i-1} = c, \bar{x}_i) \\ P(S_i = 0 | S_{i-1} = 0, X_i) &= 1 \text{ for } X_i \in \{x_i, \bar{x}_i\}. \end{aligned}$$

Essas regras garantem coerência nas probabilidades. Note que incluímos um nó extra S_0 com $P(S_0 = c) = \frac{1}{m+1}$ para todo c .

Considere o cálculo de $P(S_n = c)$ para $c \neq 0$. Perceba que

$$P(S_n = c) = \sum_{X_n \in \{x_n, \bar{x}_n\}} P(S_n = c | S_{n-1} = c, X_n) P(X_n) P(S_{n-1} = c)$$

pois, para $c \neq 0$, toda vez que $\{S_i = c\}$ e $\{S_{i-1} \neq c\}$ aparecem juntos na fórmula o termo é levado a zero.

Assim $P(S_{n-1} = c)$ não depende de X_n (ainda assumindo $c \neq 0$), e temos $P(S_n = c) = P(S_{n-1} = c) A_{c,n}$, onde $A_{c,i}$, para $i \in \{1, \dots, n\}$, é definido como segue:

$$\begin{aligned} A_{c,i} &= P(S_i = c | S_{i-1} = c) \\ &= \sum_{X_i \in \{x_i, \bar{x}_i\}} P(S_i = c | S_{i-1} = c, X_i) P(X_i). \end{aligned}$$

Podemos alterar a ordem dos cálculos, obtendo:

$$\begin{aligned}
P(S_n = c) &= P(S_{n-1} = c) A_{c,n} \\
&= \left[\sum_{X_{n-1}} P(S_{n-1} = c | S_{n-2} = c, X_{n-1}) P(X_{n-1}) P(S_{n-2} = c) \right] A_{c,n} \\
&= P(S_{n-2} = c) \left[\sum_{X_{n-1}} P(S_{n-1} = c | S_{n-2} = c, X_{n-1}) P(X_{n-1}) \right] A_{c,n} \\
&= P(S_{n-2} = c) A_{c,n-1} A_{c,n} \\
&\dots \\
&= P(S_0 = c) \prod_{i \in \{1, \dots, n\}} A_{c,i} \\
&= \frac{1}{m+1} \prod_{i \in \{1, \dots, n\}} A_{c,i}.
\end{aligned}$$

$A_{c,i}$ pode assumir três valores, dependendo da influência de X_i na cláusula C_c :

- se x_i e \bar{x}_i não aparecem em C_c , então $A_{c,i} = 1$.
- se x_i aparece em C_c e $P(x_i) = 1 - \varepsilon$, então $A_{c,i} = \varepsilon$.
- se x_i aparece em C_c e $P(x_i) = \varepsilon$, então $A_{c,i} = 1 - \varepsilon$.
- se \bar{x}_i aparece em C_c e $P(\bar{x}_i) = 1 - \varepsilon$, então $A_{c,i} = \varepsilon$.
- se \bar{x}_i aparece em C_c e $P(\bar{x}_i) = \varepsilon$, então $A_{c,i} = 1 - \varepsilon$.

Podemos concluir que se $(m+1)P(S_n = c) \leq \alpha = (1 - \varepsilon)^2 \varepsilon$, então algum X_i satisfaz C_c .

Além disso, sabemos que se C_c não foi satisfeita, então $(m+1)P(S_n = c) = \beta = (1 - \varepsilon)^3$.

Note que, com $0 < \varepsilon < \frac{1}{m+1}$, temos $\beta > m\alpha$.

$$\beta = (1 - \varepsilon)^3 > \left(1 - \frac{1}{m+1}\right)^3 = \left(\frac{m}{m+1}\right)^3 = \left(\frac{m}{m+1}\right)^2 \frac{m}{m+1} = m(1 - \varepsilon)^2 \varepsilon = m\alpha.$$

Para encontrar quantas cláusulas não foram satisfeitas, temos que somar sobre todas as categorias de S_n exceto a zero, obtendo

$$P(S_n = 0) = 1 - \sum_{c \in \{1, \dots, m\}} P(S_n = c)$$

e assim $\overline{P}(S_n = 0)$ minimiza essa soma:

$$\overline{P}(S_n = 0) = 1 - \min \left(\sum_{c \in \{1, \dots, m\}} P(S_n = c) \right).$$

Agora definimos

$$r = 1 - \frac{\beta(m - k)}{m + 1}$$

e então calculamos se $\overline{P}(S_n = 0) > r$. Temos que

$$\begin{aligned} \overline{P}(S_n = 0) &> r \iff \\ \overline{P}(S_n = 0) &> 1 - \frac{\beta(m - k)}{m + 1} \iff \\ (m + 1)\overline{P}(S_n = 0) &> m + 1 + \beta(k - m) \iff \\ \frac{(m + 1)(1 - \overline{P}(S_n = 0))}{\beta} &> k - m \iff \\ m - \frac{(m + 1)(\min \sum_{c \in \{1, \dots, m\}} P(S_n = c))}{\beta} &> k \implies \\ m - \left\lfloor \frac{(m + 1)(\min \sum_{c \in \{1, \dots, m\}} P(S_n = c))}{\beta} \right\rfloor &> k. \end{aligned}$$

Como $N = \lfloor \frac{(m+1)(\min \sum_{c \in \{1, \dots, m\}} P(S_n = c))}{\beta} \rfloor$ é exatamente o número de cláusulas não satisfeitas, estamos calculando exatamente se $m - N$ (número de cláusulas satisfeitas) é maior que k .

Assim a inferência PT-CN-Pr resolve o problema MAX-3-SAT. \square

Observação A prova continua válida se substituirmos todos os ε por zero; ela torna-se mais simples mas passa a depender de eventos de probabilidade zero, o que pode ser inconveniente, como apontado por (ZAFFALON; FAGIOLI, 2003). Note também que a prova pode ser trivialmente reescrita usando inequações no lugar de vértices, já que todos os conjuntos credais utilizados aparecem em nós binários da rede (a pertinência e a dificuldade em NP continuam valendo).

Corolário 3.23 BIW-CN-Pr é NP-Completo.

Prova A dificuldade é imediata porque PT-CN-Pr (uma especialização de BIW-CN-Pr) é NP-Completo (Teorema 3.21). A pertinência é trivial também, já que os dados vértices de cada conjunto credal, obtemos um problema BIW-BN-Pr, que pode ser resolvido em tempo polinomial (Teorema 3.3). \square

Corolário 3.24 *Não existe esquema de aproximação polinomial (PTAS) para PT-CN-Pr (ou BIW-CN-Pr) a menos que $P=NP$.*

Prova Segue diretamente do fato que MAX-3-SAT não tem um PTAS a menos que $P=NP$. Isso implica que PT-CN-Pr e BIW-CN-Pr são MAXSNP-difíceis. \square

Teorema 3.25 *CN-Pr é NP^{PP} -Completo.*

Prova Pertinência em NP^{PP} é imediata, pois dado um vértice para cada conjunto credal o problema reduz-se a BN-Pr, que é PP-Completo.

Para mostrar que é NP^{PP} -difícil, reduzimos E-MAJSAT ao nosso problema, de maneira semelhante a (PARK; DARWICHE, 2004).

Definição 3.26 *Dada uma fórmula booleana ϕ sobre um conjunto de variáveis $\{X_1, \dots, X_n\}$, e um inteiro $1 \leq k \leq n$, E-MAJSAT consiste em decidir se existe uma instânciação para as variáveis X_1, \dots, X_k tal que a maioria das possíveis instânciações de X_{k+1}, \dots, X_n satisfaça ϕ .*

Para esta redução, construímos uma rede credal modelando ϕ . Esta rede tem um nó binário para cada variável X_i , sem pais e com distribuição a priori uniforme e um nó binário para cada operador \wedge, \vee e \neg da fórmula, com categorias **true** e **false**. Os pais de um nó operador γ são seus operandos na fórmula e $P(\gamma|pa(\gamma))$ é definida pela tabela verdade do operador. Além disso, incluímos um filho X'_i para cada X_i , $i \leq k$, com probabilidades entre $[0, 1]$, respeitando $\sum_{X'_i} P(x'_i|X_i) = 1$. Seja β o único operador sem filhos na rede. Note que, para uma instânciação particular x das variáveis X_1, \dots, X_k , temos que

$$P(x|\beta = \mathbf{true}) = \sum_y P(x, y|\beta = \mathbf{true})$$

onde y é uma instânciação para X_{k+1}, \dots, X_n . Assim temos uma soma sobre todos possíveis valores de y . Isto representa o número de instânciações que satisfazem ϕ dividido por $\frac{1}{2^{n-k}}$.

Como existem $\frac{1}{2^{n-k}}$ escolhas diferentes para y (e desejamos encontrar a maioria delas), é necessário procurar por um x que faça esse número maior que $\frac{1}{2^{k+1}}$. Isso pode ser

alcançado com

$$\begin{aligned} \overline{P}(x'_1, \dots, x'_k | \beta = \mathbf{true}) &= \max \frac{P(x'_1, \dots, x'_k, \beta = \mathbf{true})}{P(\beta = \mathbf{true})} \\ &= \frac{\overline{P}(\beta = \mathbf{true} | x'_1, \dots, x'_k) \frac{1}{2^k}}{\frac{1}{2^n}} \\ &= 2^{n-k} \overline{P}(\beta = \mathbf{true} | x'_1, \dots, x'_k) \end{aligned}$$

que é exatamente o cálculo de MAP feito por (PARK; DARWICHE, 2004). Assim, se e somente se $\overline{P}(\beta = \mathbf{true} | x'_1, \dots, x'_k) > \frac{1}{2^{n+1}}$, temos uma instanciação (dada pelos vértices de $P(x'_i | X_i)$) que garante a maioria. \square

É conhecido que o problema CN-Pr pode ser resolvido pelo problema BN-MAP, conduzindo uma transformação *CCM* na rede credal (COZMAN, 2000a). O Teorema 3.27 apresenta o resultado reverso entre inferências em redes Bayesianas e credais.

Teorema 3.27 *O problema BN-MAP pode ser reduzido ao problema CN-Pr com consultas sobre várias variáveis sem alteração no nível de dificuldade (PT, BIW, ou multi-conectada) da topologia da rede nem dos tipos de nós (binários, ternários. etc).*

Prova Suponha que X_1, \dots, X_n são as variáveis de MAP. Adicione à rede um filho binário X'_i para cada X_i , com $P(X'_i | X_i) \in [0, 1]$ e a restrição $\sum_{X'_i} P(x'_i | X_i) = 1$. Assim temos

$$\max_{X_1, \dots, X_n} P(X_1, \dots, X_n | \mathbb{E}) = \overline{P}(x'_1, \dots, x'_n | \mathbb{E})$$

Após calcular $\overline{P}(x'_1, \dots, x'_n | \mathbb{E})$, precisamos apenas olhar para cada nó X'_i e escolher a categoria de X_i de acordo com qual das probabilidades $P(x'_i | X_i)$ é igual a um (exatamente uma será). Assim, decidir se $\overline{P}(x'_1, \dots, x'_n | \mathbb{E}) > r$ é o mesmo que decidir se existe uma instanciação para as variáveis de MAP que tenha probabilidade maior que r . \square

Diversos problemas práticos necessitam de consultas sobre várias variáveis no lugar de apenas uma. Os Lemas 3.29 e 3.28 afirmam que a dificuldade dessas consultas conjuntas não é maior que consultas sobre uma variável. Note que consultas sobre uma ou várias variáveis em redes Bayesianas são diretamente relacionadas. Infelizmente isso não acontece nas redes credais (pois várias consultas sobre uma variável não são capazes de resolver um problema sobre várias variáveis), mas a classe de complexidade dos problemas continua a mesma.

Lema 3.28 *CN-Pr com consulta sobre várias variáveis é NP^{PP} -Completo.*

Prova A dificuldade é imediata, pois CN-Pr com consulta sobre uma variável já é NP^{PP}-difícil. A pertinência em NP^{PP} também é trivial, pois dado um vértice para cada conjunto credal da rede, o problema reduz-se a BN-Pr com consulta sobre várias variáveis, que é PP-Completo. \square

Lema 3.29 BIW-CN-Pr com consulta sobre várias variáveis é NP-Completo.

Prova A dificuldade é imediata, pois BIW-CN-Pr com consulta sobre uma variável já é NP-difícil. A pertinência é obtida pois, escolhendo um vértice para cada conjunto credal, o problema reduz-se a BIW-BN-Pr com consulta sobre várias variáveis, que é polinomial: suponha que as variáveis de consultas são x_1, \dots, x_k e e é a evidência (observação). Então

$$P(x_1, \dots, x_k | \mathbb{E}) = \prod_i P(x_i | x_{i+1}, \dots, x_k, \mathbb{E})$$

Como cada $P(x_i | x_{i+1}, \dots, x_k, \mathbb{E})$ pode ser calculado separadamente, k consultas sobre uma variável são capazes de resolver o problema sobre várias variáveis na rede Bayesiana. \square

3.3.2 Explicação mais provável

Definimos inicialmente versões *maximin* dos problemas MPE em redes credais.

Definição 3.30 Dada uma rede credal $(G, \mathbb{X}, \mathbb{K})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, e um número racional r , CN-MPEe é a questão de decidir se existe uma instânciação q para as variáveis de $\mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$ tal que $\underline{P}(q | \mathbb{E}) > r$.

Teorema 3.31 PT-CN-MPEe e BIW-CN-MPEe são Σ_2^P -Completos.

Prova Para mostrar a pertinência basta olhar para BIW-CN-MPEe, que é uma generalização de PT-CN-MPEe. BIW-CN-MPEe pertence a Σ_2^P , pois dada a instânciação q , obtemos um problema complementar a BIW-CN-Pr, que está em NP (Corolário 3.23). Assim um oráculo coNP é suficiente.

Para mostrar a dificuldade, reduzimos para PT-CN-MPEe uma versão de QSAT₂ que é Σ_2^P -Completa (PAPADIMITRIOU, 1994):

Definição 3.32 Dado um conjunto de variáveis $\mathbb{X} = \{X_1, \dots, X_n\}$, um inteiro $0 < k \leq n$ e uma fórmula booleana 3DNF $\phi(\mathbb{X})$ sobre estas variáveis, QSAT₂^{DNF} consiste em decidir se existe uma instânciação para as primeiras k variáveis tal que, para toda instânciação das demais $n - k$ variáveis, $\phi(\mathbb{X})$ é satisfeita.

Inicialmente construímos uma rede similar àquela do Teorema 3.21. Para cada variável X_i , com $1 \leq i \leq k$, construímos dois nós, ditos X_i e S_i . O primeiro é binário, não tem pais e representa o estado da variável booleana X_i ; as probabilidades $P(X_i = x_i)$ e $P(X_i = \bar{x}_i)$ estão no intervalo $[\varepsilon, 1 - \varepsilon]$ ($0 < \varepsilon < \frac{1}{m+1}$ é uma constante pequena). O último assume valores de 0 até m , ou seja, possui $m + 1$ categorias, tem S_{i-1} e X_i como pais e tem probabilidades definidas por

$$\begin{aligned} P(S_i = c | S_{i-1} = c, x_i) &= 0 \text{ se } \bar{x}_i \in C_c, \text{ ou } 1 \text{ caso contrário} \\ P(S_i = c | S_{i-1} = c, \bar{x}_i) &= 0 \text{ se } x_i \in C_c, \text{ ou } 1 \text{ caso contrário} \\ P(S_i = c | S_{i-1} \neq c, X_i) &= 0 \text{ para } X_i \in \{x_i, \bar{x}_i\}, \end{aligned}$$

para $c \neq 0$. Quando $S_i = 0$, temos as regras:

$$\begin{aligned} P(S_i = 0 | S_{i-1} = c, X_i) &= 1 - P(S_i = c | S_{i-1} = c, X_i) \\ P(S_i = 0 | S_{i-1} = c, x_i) &= 1 - P(S_i = c | S_{i-1} = c, x_i) \\ P(S_i = 0 | S_{i-1} = c, \bar{x}_i) &= 1 - P(S_i = c | S_{i-1} = c, \bar{x}_i) \\ P(S_i = 0 | S_{i-1} = 0, X_i) &= 1 \text{ for } X_i \in \{x_i, \bar{x}_i\}. \end{aligned}$$

Essas regras garantem coerência nas probabilidades. Note que incluímos um nó extra S_0 com $P(S_0 = c) = \frac{1}{m+1}$ para todo c .

As variáveis X_{k+1}, \dots, X_n são ternárias, assumindo as categorias x_i, \bar{x}_i, o_i . Suas probabilidades são: $P(X_i = o_i) = \varepsilon$ e $P(X_i = x_i), P(X_i = \bar{x}_i)$ pertencem ao intervalo $[0, 1 - \varepsilon]$, onde $0 < \varepsilon < \frac{1}{m+2}$ é uma constante. As probabilidades de S_i são definidas por $P(S_i = c | S_{i-1} = c, o_i) = 0$, for $c \neq 0$ (o caso onde $c = 0$ continua igual, isto é, igual a 1).

Além disso, incluímos um nó extra Q que tem S_n como seu único pai. Definimos $P(q | S_n = c) = 0$ para $c \neq 0$ e 1 caso contrário.

Iremos resolver o problema PT-CN-MPEe definido por

$$\underline{P}(X, S | q) > r = \frac{\frac{\varepsilon^n}{m+1}}{1 - \frac{m}{m+1}\varepsilon(1 - \varepsilon)^2}$$

onde $S = \{S_0, \dots, S_n\}$ e $X = \{X_1, \dots, X_n\}$. Seja X^- o conjunto $\{X_1, \dots, X_k\}$ e X^+ o conjunto $\{X_{k+1}, \dots, X_n\}$. Então

$$P(X^-, X^+, S|q) = \frac{P(q|S) P(S|X) P(X^-) P(X^+)}{P(q)}.$$

Note que $P(q|S) \neq 0$ apenas para $S_n = 0$. Além disso, existem instanciações para S que fazem $P(S|X)$ igual a zero e outras que fazem $P(S|X)$ igual a $\frac{1}{m+1}$ (perceba que a instanciação com $s' = \{\forall i S_i = 0\}$ tem probabilidade igual a $\frac{1}{m+1}$).

Assim sabemos que a solução do problema PT-CN-MPEe será encontrada em uma instanciação onde $P(S|X) = \frac{1}{m+1}$. Temos também que a escolha da categoria o_i para todas as variáveis X^+ leva-nos a probabilidades diferentes de zero. Isto é.

$$\exists(X^-, X^+, S) \underline{P}(X^-, X^+, S|q) > 0 \iff \exists(X^-) \underline{P}(X^-, o, s|q) > 0,$$

onde o denota $\{X_i = o_i \text{ para } i \in \{k+1, \dots, n\}\}$ e s denota uma instanciação tal que $P(S|X) = \frac{1}{m+1}$. Se escolhermos uma categoria diferente de o_i em qualquer um dos lugares, a probabilidade máxima seria zero, já que $P(X_i \neq o_i) \in [0, 1 - \varepsilon]$, e assim essa probabilidade poderia assumir valor zero na minimização interna.

$$P(X^-, o, s|q) = \frac{\frac{1}{m+1} \varepsilon^{n-k} P(X^-)}{1 - \sum_{c \neq 0} P(S_n = c)}.$$

Quando calculamos $\underline{P}(X^-, o, s|q)$, o numerador $P(X^-)$ torna-se ε^k , pois qualquer solução que não faça $P(X^-) = \varepsilon^k$ não será um mínimo para $\underline{P}(X^-, o, s|q)$ (lembre-se que $\varepsilon < \frac{1}{m+2}$). Isto é verdade porque apenas uma variável X^- usando o ponto extremo $(1 - \varepsilon)$ no lugar de ε é suficiente para fazer $\underline{P}(X^-, o, s|q)$ muito grande:

$$\forall R, R' \quad \frac{\frac{1}{m+1} \varepsilon^{n-k} \varepsilon^{k-1} (1 - \varepsilon)}{1 - R} > \frac{\frac{1}{m+1} \varepsilon^{n-k} \varepsilon^k}{1 - R'}$$

onde R e R' são quaisquer possíveis valores para a soma $\sum_{c \neq 0} P(S_n = c)$ que aparece no denominador (note que essas somas são restritas em $[0, \frac{m}{m+1}]$ pelas probabilidades da rede).

Resumindo, todas as variáveis S na solução de PT-CN-MPEe estarão fixadas de forma que $P(S|X) = \frac{1}{m+1}$, todas as variáveis X^+ estarão fixadas em o , e a instanciação escolhida para X^- faz $P(X^-) = \varepsilon^k$, o que implica que se x_i pertencer a instanciação de X^- , então

$P(x_i) = \varepsilon$ e $P(\bar{x}_i) = 1 - \varepsilon$ (o caso oposto é análogo). Isso significa que os vértices dos conjuntos credais dos nós X^- são completamente definidos pela instanciação escolhida (não estão livres). Assim apenas os conjuntos credais que estão livres no denominador são aquelas das variáveis X^+ (os nós das variáveis S_i não são credais). Processando o problema PT-CN-MPEe, temos que

$$\underline{P}(X^-, o, s|q) = \frac{\frac{1}{m+1}\varepsilon^n}{1 - \min\left(\sum_{c \neq 0} P(S_n = c)\right)},$$

onde $\min\sum_{c \neq 0} P(S_n = c)$ é calculado sobre todos os possíveis vértices dos conjuntos credais de X^+ (sabemos que $P(X_i = o_i)$ está fixada em ε , mas as probabilidades $P(X_i = x_i)$ e $P(X_i = \bar{x}_i)$ podem variar entre 0 e $1 - \varepsilon$).

O problema PT-CN-MPEe terá resposta positiva quando $\min\sum_{c \neq 0} P(S_n = c) > \frac{m}{m+1}\varepsilon(1 - \varepsilon)^2$. Assim, basta que $P(S_n = c) = \frac{1}{m+1}(1 - \varepsilon)^3$ para algum c . Sabemos, pelas definições das probabilidades da rede, que $P(S_n = c) = \frac{1}{m+1}(1 - \varepsilon)^3$ acontece exatamente quando a cláusula c não é negada por nenhuma de suas variáveis (ou seja, é satisfeita).

$$\min\sum_{c \neq 0} P(S_n = c) = \frac{1}{m+1} \min\sum_{c \neq 0} \prod_{i=1}^n A_{c,i}$$

onde

$$A_{c,i} = \begin{cases} P(S_i = c|S_{i-1} = c, x_i) P(x_i) + P(S_i = c|S_{i-1} = c, \bar{x}_i) P(\bar{x}_i) \\ = \begin{cases} 1 & \text{se } x_i \text{ não aparece na cláusula } c \\ P(x_i) & \text{se } x_i \in \text{cláusula } c \\ P(\bar{x}_i) & \text{se } \bar{x}_i \in \text{cláusula } c \\ 0 & \text{se } x_i \text{ e } \bar{x}_i \in \text{cláusula } c \end{cases} \end{cases}$$

e $A_{c,i}$, para $i \leq k$, é definido pela escolha de X^- e $A_{c,i}$, para $i > k$, é definido pelos vértices escolhidos dos conjuntos credais durante a minimização de $P(S_n = c)$. $A_{c,i}$ é igual a 1 se a variável i não aparece na cláusula c , $(1 - \varepsilon)$ se a variável i não nega a cláusula c e $(0$ ou $\varepsilon)$ se i nega a cláusula. Dessa forma, se $\min\sum_{c \neq 0} P(S_n = c) > \frac{m}{m+1}\varepsilon(1 - \varepsilon)^2$ então para todas as instanciações de X^+ , a fórmula $\phi(X)$ é satisfeita. Esta é exatamente a consulta necessária para QSAT_2^{DNF} . \square

Corolário 3.33 CN-MPEe é PP-difícil e Σ_2^P -difícil.

Prova Segue diretamente do fato de CN-MPEe ser uma generalização de BN-MPEe, que é PP-Completo, e de PT-CN-MPEe, que é Σ_2^P -Completo. \square

Vamos olhar agora para o problema CN-MPE. Ele é semelhante ao problema CN-MPEe, mas sem considerar a probabilidade $P(\mathbb{E})$ da evidência no cálculo.

Definição 3.34 *Dada uma rede credal $(G, \mathbb{X}, \mathbb{K})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, e um número racional r , CN-MPE é a questão de decidir se existe uma instanciação q para as variáveis de $\mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$ tal que $\underline{P}(q, \mathbb{E}) > r$.*

No contexto de redes Bayesianas, a diferença entre MPE e MPEe pode ser considerada superficial, já que a instanciação que leva à explicação mais provável pode ser encontrada com o problema BN-MPE. Todavia o mesmo não acontece para redes credais, onde não é possível encontrar a explicação mais provável usando MPE, sem utilizar a evidência nos cálculos (note que CN-MPE e CN-MPEe possuem diferenças não triviais).

Teorema 3.35 *PT-CN-MPE e BIW-CN-MPE estão em P.*

Prova É suficiente perceber que a minimização interna da consulta do problema BIW-CN-MPE

$$\max_x \min_{P \in K(\mathbb{X})} \prod_i P(x_i | \text{pa}(x_i))$$

fatora-se, já que a rede é localmente especificada (note que os x_i 's devem ser consistentes com a observação e). A otimização torna-se

$$\max_x \prod_i \underline{P}(x_i | \text{pa}(x_i))$$

que é equivalente à otimização do problema BIW-BN-MPE. Dessa forma, uma simples modificação do algoritmo **elim-max** (DECHTER, 1996) resolve o problema: no lugar de usar $P(x_p | \text{pa}(x_p))$ no algoritmo **elim-max**, usamos $\underline{P}(x_p | \text{pa}(x_p))$ (sempre que necessário). O resto do procedimento continua inalterado. Essa modificação resolve o problema pois cada conjunto credal é localmente especificado. Quando processamos o *bucket* p , **elim-max** faz

$$h_p(u) = \max_{x_p} \prod_{i=1}^j h_i(x_p, u_{S_i})$$

onde $h_i(x_p, u_{S_i})$ são todas as funções relacionadas a X_p , e então passa essa nova função para os próximos *buckets*. (DECHTER, 1996) chama cada $P(x_p | \text{pa}(x_p))$ de $h_i(x_p, u_{S_i})$. O que fazemos é resolver o problema de minimização localmente em cada $K(x_p | \text{pa}(x_p))$, encontrando $\underline{P}(x_p | \text{pa}(x_p))$, e então usamos esse valor no lugar de $P(x_p | \text{pa}(x_p))$. Essas minimizações pode ser feitas separadamente, pois o MPE escolhe apenas um dentre todos os $\underline{P}(x_p | \text{pa}(x_p))$ de um dado nó. \square

Teorema 3.36 *CN-MPE é NP-Completo.*

Prova O caráter NP-difícil é imediato, pois o problema BN-MPE já é NP-Completo e pode ser trivialmente transformado em CN-MPE (temos apenas que usar uma rede credal composta por conjuntos com uma única densidade de probabilidade). Pertinência é alcançada pois, dada uma instanciãção x para as variáveis, o valor de $\underline{P}(x)$ é dado por $\prod_i \underline{P}(x_i | \text{pa}(x_i))$. Isso vale pois cada conjunto credal $K(x_i | \text{pa}(x_i))$ é localmente especificado. \square

3.3.3 Máximo a posteriori

Definimos aqui a versão *maximin* do problema MAP em redes credais.

Definição 3.37 *Dada uma rede credal $(G, \mathbb{X}, \mathbb{K})$, a evidência \mathbb{E} com $\mathbb{X}_{\mathbb{E}} \subseteq \mathbb{X}$, um conjunto de consulta $Q \subseteq \mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$ e um número racional r , CN-MAP é a questão de decidir se existe uma instanciãção q para as variáveis de Q tal que $\underline{P}(q | \mathbb{E}) > r$.*

Note que o problema CN-MPEe analisado na Seção 3.3.2 é o problema CN-MAP quando todas as variáveis participam da consulta, isto é, $Q = \mathbb{X} \setminus \mathbb{X}_{\mathbb{E}}$.

Corolário 3.38 *PT-CN-MAP e BIW-CN-MAP são Σ_2^P -Completo.*

Prova Para mostrar a pertinência basta olhar para BIW-CN-MAP, que é uma generalização de PT-CN-MAP. BIW-CN-MAP pertence a Σ_2^P , pois dada a instanciãção q , obtemos um problema complementar a BIW-CN-Pr, que está em NP (Corolário 3.23). Dessa forma um oráculo coNP é suficiente.

Para mostrar a dificuldade, basta notar que PT-CN-MAP é uma generalização do problema PT-CN-MPEe, que é Σ_2^P -Completo (Teorema 3.31). \square

Corolário 3.39 *CN-MAP é NP^{PP} -difícil.*

Prova Segue diretamente do fato de CN-MAP ser uma generalização de BN-MAP, que é NP^{PP} -Completo (Teorema 3.14). \square

3.4 Redes semi-qualitativas

Definimos aqui os problemas de decisão para redes semi-qualitativas. Uma inferência tradicional em SQPN refere-se a questão qualitativa de como a observação de algumas

SQPN-M	SQPN-m	Resposta da inferência
Sim	Sim	Influência ambígua
Sim	Não	Influência positiva
Não	Sim	Influência negativa
Não	Não	Influência zero

Tabela 3: Resposta da inferência em uma SQPN de acordo com os resultados dos problemas SQPN-M e SQPN-m.

variáveis altera a probabilidade de outras. Seja Q a variável de consulta e e a observação (um evento observado); precisamos calcular $P(q|e) - P(q)$. Quando $\max(P(q|e) - P(q)) \leq 0$, temos uma influência negativa da observação sobre Q , pois a observação de e fez diminuir o valor da probabilidade de q (a diferença $P(q|e) - P(q)$ é sempre menor ou igual a zero). Se $\min(P(q|e) - P(q)) \geq 0$, então temos uma influência positiva, pois a observação fez aumentar a probabilidade de q . Se ambos \max e \min são zero, então não temos influência. Caso contrário temos uma influência ambígua da observação na variável de consulta.

Para calcular a inferência sobre a influência de observações em uma variável de consulta, utilizamos os problemas de decisão SQPN-M e SQPN-m.

Definição 3.40 SQPN-M é o problema de decidir se existem atribuições de probabilidade para todas as configurações dos nós e seus pais (ou seja, para todos os valores de probabilidade que já não estavam previamente fixados) que façam $P(q|\mathbb{E}) - P(q) > 0$, onde Q é a variável de consulta e \mathbb{E} é nossa observação na rede.

Definição 3.41 SQPN-m é o problema de decidir se existem atribuições de probabilidade para todas as configurações dos nós e seus pais (ou seja, para todos os valores de probabilidade que já não estavam previamente fixados) que façam $P(q|\mathbb{E}) - P(q) < 0$, onde Q é a variável de consulta e \mathbb{E} é nossa observação na rede.

Claramente precisamos (e é suficiente) resolver ambos os problemas SQPN-M e SQPN-m para calcular a inferência tradicional em uma rede SQPN. A tabela 3 mostra essa relação.

Como SQPNs oferecem uma combinação de QPNs e redes Bayesianas, poderíamos esperar que a complexidade de tempo de inferências em SQPNs não seria maior que a complexidade de redes Bayesianas. Esta seção caracteriza um retrato diferente: SQPNs são mais difíceis que QPNs e redes Bayesianas.

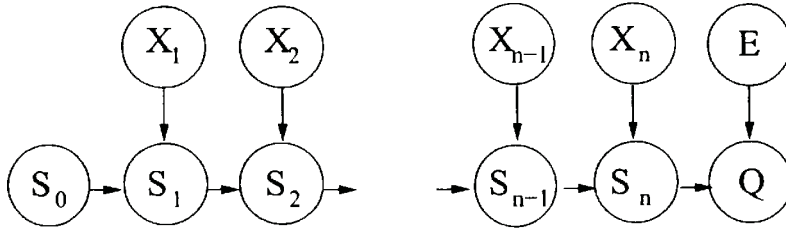


Figura 10: Rede utilizada no Teorema 3.42.

Teorema 3.42 PT-SQPN-M, BIW-SQPN-M, PT-SQPN-m e BIW-SQPN-m são NP-Completo.

Prova Primeiro note que estes problemas pertencem a NP, pois dadas as probabilidades para todas as configurações de nós e seus pais, obtemos problemas em redes Bayesianas tradicionais. Neste caso o cálculo de $\max P(q|\mathbb{E}) - P(q)$ (ou $\min P(q|\mathbb{E}) - P(q)$) pode ser feito em tempo polinomial (Corolário 3.3).

Para mostrar a dificuldade, vamos reduzir o problema MAX-3-SAT para uma consulta em PT-SQPN-M de forma análoga ao Teorema 3.21. Reproduzimos o problema por conveniência:

Dado um conjunto de variáveis booleanas $\{X_1, \dots, X_n\}$, uma fórmula 3CNF com cláusulas $\{C_1, \dots, C_m\}$ e um inteiro $0 \leq k < m$, o problema MAX-3-SAT é a questão de decidir se existe uma instância para as variáveis que satisfaz mais que k cláusulas da fórmula.

Utilizamos uma rede com topologia apresentada na Figura 10. Ela tem apenas dois nós binários adicionais com relação àquela do Teorema 3.21: E , sem pais e com probabilidade uniforme e Q com pais S_n , E e probabilidades condicionais definidas por:

$$P(q|S_n = 0, e) = 1$$

$$P(q|S_n = 0, \bar{e}) = r$$

$$P(q|S_n \neq 0, e) = 0$$

$$P(q|s_n \neq 0, \bar{e}) = r$$

(lembre que $r = 1 - \frac{\beta(m-k)}{m+1}$). As demais variáveis qualitativas são X_1, \dots, X_n . Note que neste caso elas funcionam exatamente igual às variáveis credais.

Agora uma inferência em SQPN sobre a influência de $\mathbb{E} = \{E = e\}$ sobre Q irá resolver MAX-3-SAT. Para resolver essa inferência, precisamos encontrar o sinal de $\min P(q|\mathbb{E}) - P(q)$ e $\max P(q|\mathbb{E}) - P(q)$. Vamos focar inicialmente a atenção para o

cálculo de $\max P(q|\mathbb{E}) - P(q)$. Note que

$$P(q|\mathbb{E}) = P(q|S_n = 0, e) P(S_n = 0) + \sum_{c \neq 0} P(q|S_n = c, e) P(S_n = c) = P(S_n = 0)$$

e

$$\begin{aligned} P(q) &= \left(P(q|S_n = 0, e) P(S_n = 0) + \sum_{c \neq 0} P(q|S_n = c, e) P(S_n = c) \right) P(e) + \\ &\quad + \left(P(q|S_n = 0, \bar{e}) P(S_n = 0) + \sum_{c \neq 0} P(q|S_n = c, \bar{e}) P(S_n = c) \right) P(\bar{e}) \\ &= \frac{P(S_n = 0) + r}{2}. \end{aligned}$$

Dessa forma temos que

$$P(q|\mathbb{E}) - P(q) = \frac{P(S_n = 0) - r}{2}.$$

A consulta PT-SQPN-M decide se existe alguma instância tal que esse valor é maior que zero. Isso acontece exatamente quando $\max P(S_n = 0) > r$, que é a consulta utilizada no Teorema 3.21. O restante da prova é idêntico àquele teorema. A prova para PT-SQPN-m é análoga. Alteramos ligeiramente as probabilidades condicionais em Q :

$$P(q|S_n = 0, e) = r$$

$$P(q|S_n = 0, \bar{e}) = 1$$

$$P(q|S_n \neq 0, e) = r$$

$$P(q|s_n \neq 0, \bar{e}) = 0$$

e obtemos que $P(q|\mathbb{E}) - P(q) = \frac{r - P(S_n = 0)}{2}$. Agora o argumento vale pois a consulta a PT-SQPN-m decide se esta equação pode resultar em valor negativo. Essa inferência é o mesmo que perguntar se $\max P(S_n = 0) > r$ (e chegamos novamente na consulta usada no Teorema 3.21).

Finalmente, como BIW-SQPN-M e BIW-SQPN-m são generalizações dos problemas anteriores, eles também são NP-difíceis. \square

Corolário 3.43 *Não existe esquema de aproximação polinomial (PTAS) para PT-SQPN-M, PT-SQPN-m, BIW-SQPN-M ou BIW-SQPN-m a menos que $P=NP$.*

Prova Segue diretamente do fato que MAX-3-SAT não tem um PTAS a menos que $P=NP$. Isso implica que estes problemas são MAXSNP-difíceis. \square

Teorema 3.44 *SQPN-M e SQPN-m são NP^{PP} -Completos.*

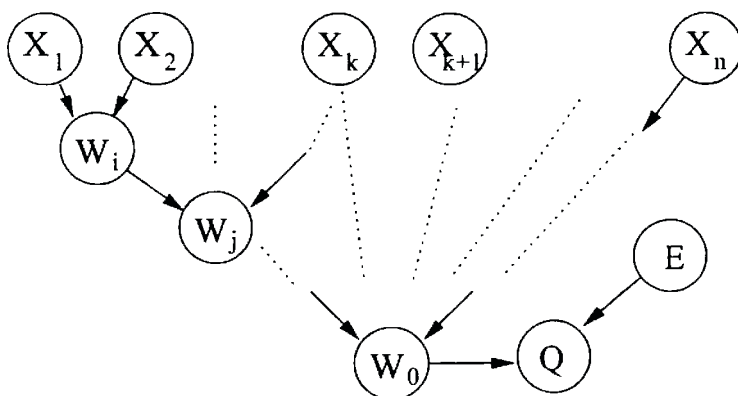


Figura 11: Rede utilizada no Teorema 3.44.

Prova Primeiro note que tanto SQPN-M quanto SQPN-m pertencem a NP^{PP} , pois dadas as probabilidades para todas as configurações de nós e seus pais, obtemos problemas em redes Bayesianas tradicionais. Neste caso o cálculo de $\max P(q|\mathbb{E}) - P(q)$ (ou $\min P(q|\mathbb{E}) - P(q)$) pode ser feito pelo oráculo PP.

Para mostrar a dificuldade dos problemas, reduzimos o problema E-MAJSAT, que descrevemos novamente por conveniência:

Dada uma fórmula booleana ϕ sobre as variáveis $\mathbb{X} = \{X_1, \dots, X_n\}$, e um inteiro $1 \leq k \leq n$, existe uma instanciação para as variáveis X_1, \dots, X_k tal que a maioria das instancicações de \mathbb{X} satisfazem ϕ ?

Seja \mathbb{X} o conjunto das primeiras k variáveis e \mathbb{Y} as demais, isto é, X_{k+1}, \dots, X_n . Construimos uma SQPN modelando a fórmula ϕ . Esta rede tem um nó qualitativo para cada variável em \mathbb{X} , sem pais. As variáveis de \mathbb{Y} não têm pais e possuem probabilidade a priori uniforme. Além disso, existe um nó W_i para cada operador booleano. Os pais de um operador W_i são seus operandos na fórmula e $P(w_i|\text{pa}(W_i))$ descreve sua tabela verdade. Seja W_0 o único operador sem filhos na rede. Insira um nó extra a ele, chamado de Q , com W_0 e um novo nó qualitativo E como seus pais (veja Figura 11). Fixamos

$$\begin{aligned} P(q|w_0, e) &= \frac{1}{2} \\ P(q|w_0, \bar{e}) &= 1 \\ P(q|\bar{w}_0, e) &= \frac{1}{2} \\ P(q|\bar{w}_0, \bar{e}) &= 0. \end{aligned}$$

Agora uma inferência em SQPN sobre a influência de $\mathbb{E} = \{E = e\}$ sobre Q irá resolver E-MAJSAT. Para resolver essa inferência, precisamos encontrar o sinal de $\min P(q|\mathbb{E}) - P(q)$ e $\max P(q|\mathbb{E}) - P(q)$. Vamos focar inicialmente a atenção para o cálculo de $\min P(q|\mathbb{E}) - P(q)$. Note que

$$P(q|\mathbb{E}) = P(q|w_0, e) P(w_0) + P(q|\bar{w}_0, e) P(\bar{w}_0) = \frac{1}{2}$$

e

$$\begin{aligned} P(q) &= (P(q|w_0, e) P(w_0) + P(q|\bar{w}_0, e) P(\bar{w}_0)) P(e) + \\ &\quad + (P(q|w_0, \bar{e}) P(w_0) + P(q|\bar{w}_0, \bar{e}) P(\bar{w}_0)) P(\bar{e}) \\ &= P(w_0) (1 - P(e)) + \frac{P(e)}{2}. \end{aligned}$$

Assim $P(w_0) > \frac{1}{2}$ implica em $\min P(q|\mathbb{E}) - P(q) < 0$ e $P(w_0) \leq \frac{1}{2}$ implica em $\min P(q|\mathbb{E}) - P(q) = 0$ (pois $P(e)$ será fixado em 1 neste caso). Suponha que $P(w_0)$ indique o número de instâncias de \mathbb{Y} que satisfazem ϕ ; então precisamos calcular apenas a resposta da consulta qualitativa: se $\min P(q|\mathbb{E}) - P(q) < 0$, encontramos uma instância de \mathbb{X} que satisfaz os requisitos do problema E-MAJSAT, ou seja, onde a maioria das instâncias de \mathbb{Y} satisfazem ϕ . Caso contrário não existe tal instância para as variáveis \mathbb{X} . Ainda é necessário mostrar que $P(w_0)$ indica o número de instâncias de \mathbb{Y} satisfazendo ϕ , dado a instância de \mathbb{X} (então a consulta $\min \frac{1}{2} - P(q)$ irá maximizar $P(w_0)$, encontrando a instância de \mathbb{X} desejada). Temos

$$\begin{aligned} P(w_0) &= \sum_{X,Y} P(w_0|X, Y) P(X) P(Y) \\ &= \sum_Y \left(P(w_0|xY) \frac{1}{2^{n-k}} \right) = \frac{|\text{sat}|}{|\text{total cases}|}. \end{aligned}$$

Note que o somatório sobre X apresentado desaparece, pois existe apenas um termo onde $P(X)$ é igual a 1 (e $P(X) = 0$ nos outros casos). Chamamos aquela instância não zerada de $\{X = x\}$. Além disso, $P(w_0|xY)$ é 1 quando (x, Y) satisfaz ϕ , e 0 caso contrário. Isso implica que $\max P(w_0)$ calcula qual instância de \mathbb{X} tem a maior proporção de instâncias de \mathbb{Y} satisfazendo ϕ . Os mesmos argumentos valem para SQPN-M. \square

Utilizamos no Teorema 3.44 uma rede com relações qualitativas muito simples, e obtivemos que os problemas são NP^{PP}-difíceis; a inclusão de outras relações qualitativas, como influências e sinergias, sinais situacionais e relações não monotônicas apenas tornam o pro-

blema mais difícil, mas não o fazem sair de NP^{PP} , pois fixados os valores de probabilidade em todos os nós. continuamos obtendo uma inferência de atualização de crenças em uma rede Bayesiana. Isso implica que inferências *exatas* em redes qualitativas especializadas (PARSONS; DOHNAL, 1993; RENOIJ; GAAG, 2002) também são NP^{PP} -Completas.

4 *Inferências usando programação multilinear*

4.1 Introdução

Neste capítulo tratamos como representar e resolver inferências em redes credais e qualitativas utilizando programação multilinear. Apresentamos ainda extensões e outros problemas que podem ser beneficiados por um algoritmo eficiente para programação multilinear. Na Seção 4.2 introduzimos as idéias e estruturas dos problemas de programação multilinear, apresentando na Seção 4.3 um algoritmo para sua resolução. Apresentamos então as inferências em redes credais (Seção 4.4) e como transformá-las em problemas de programação multilinear (Seção 4.4.1). Finalmente, na Seção 4.5, tratamos das inferências em redes semi-qualitativas.

Na Seção 4.6 tratamos de algoritmos aproximados para o problema. Mostramos como entender o algoritmo A/R+ de Rocha et al (ROCHA; COZMAN; CAMPOS, 2003) para redes multi-conectadas (Seção 4.6.1) e o algoritmo A/R++ (CAMPOS; COZMAN, 2004; COZMAN; CAMPOS, 2004), criado a partir das idéias do algoritmo A/R+ com a introdução de programação multilinear (Seção 4.6.2). Na Seção 4.6.3 discutimos o algoritmo ILS (*busca local iterativa*), baseado na idéia de (LUKATSKII; SHAPOT, 2000) para problemas multilineares.

Finalizando o capítulo, apresentamos na Seção 4.7.1 os resultados obtidos com o algoritmo ILS e na Seção 4.7.2 experimentos realizados com os algoritmos RL (para programação multilinear) e A/R++.

4.2 Programação multilinear

Nesta seção discutimos como resolver problemas de programação multilinear. Embora o uso de programação multilinear para inferências em redes credais e qualitativas tenha sido mencionado na literatura (ANDERSEN; HOOKER, 1994; COZMAN, 2000a; ZAFFALON, 1997), aparentemente nenhum estudo sistemático dessas propostas foi conduzido até o momento, e nenhuma implementação para programação multilinear voltada a inferências em redes credais é conhecida. Apresentamos um algoritmo para tal de forma detalhada, com aplicação direta em inferências.

Um problema de programação multilinear busca por um máximo (ou mínimo) da função multilinear objetivo restringida a um conjunto de funções multilineares, definidas em termos de variáveis contínuas. Uma formulação para o problema segue.

$$\text{MP}(\Omega) : \text{maximizar } \{\phi_0(x) : x \in Z \cap \Omega\}$$

onde

$$\begin{aligned} Z &= \{x : \phi_r(x) \geq \beta_r \text{ para } r = 1, \dots, R_1, \\ &\quad \phi_r(x) = \beta_r \text{ para } r = R_1 + 1, \dots, R\} \\ \Omega &= \{x : 0 \leq l_j \leq x_j \leq u_j \leq 1, \text{ para } j = 1, \dots, n\} \end{aligned}$$

e onde

$$\phi_r(x) = \sum_{t \in T_r} \alpha_{rt} \left[\prod_{j \in J_{rt}} x_j \right] \text{ para } r = 0, \dots, R$$

Aqui, T_r é um conjunto de índices para os termos definindo ϕ_r e α_{rt} são os coeficientes racionais para os termos multilineares $(\prod_{j \in J_{rt}} x_j)$, $t \in T_r$, $r = 0, \dots, R$. Cada conjunto J_{rt} define quais variáveis aparecem no termo t da função ϕ_r . Definimos como T o conjunto de todos os termos no problema, isto é, $T = \cup_r T_r$. Note que não permitimos repetições de índices nos conjuntos, pois o problema é multilinear. Seja $N = \{1, \dots, n\}$ e δ o grau máximo de qualquer termo aparecendo no problema $\text{MP}(\Omega)$. Assim cada $J_{rt} \subseteq N$, com $1 \leq |J_{rt}| \leq \delta$, para $t \in T_r$, $r = 0, \dots, R$.

Problemas desse tipo têm uma ampla área de aplicação. Qualquer problema multilinear pode ser escrito desta forma apresentada. Diferentemente de problemas de programação geométrica (AVRIEL, 1980) (onde todos α_{rt} tem o mesmo sinal e os expoentes são li-

vres), problemas do tipo MP são não convexos e nenhuma transformação conhecida pode convexificá-los: eles podem ter múltiplos ótimos locais e regiões viáveis não convexas.

Diversos métodos de otimização local são aplicáveis para MP: procedimentos baseados em condensação (AVRIEL; DEMBO; PASSY, 1975; DUFFIN, 1970), condições de Karush-Kuhn-Tucker (RIJCKAERT; MARTENS, 1978), técnicas de linearização (LUKATSKII; SHAPOT, 2000), e métodos de programação não linear genéricos (HAARHOFF; BUYS, 1970; PARDALOS; ROSEN, 1987; RATNER; LASDON; JAIN, 1978). Os algoritmos exatos existentes mais comuns são baseados em técnicas de *branch-and-bound* (GOCHET; SMEERS, 1979; MARANAS; FLOUDAS, 1997; SHERALI; TUNCBILEK, 1992) ou métodos de planos de corte (HORST; TUY, 1996; TUY, 1998). As idéias de *branch-and-bound* apresentadas em (MARANAS; FLOUDAS, 1997) e (GOCHET; SMEERS, 1979) são baseadas na resolução de sub-problemas convexos não lineares, enquanto a idéia de (SHERALI; TUNCBILEK, 1992) é baseada em sub-problemas lineares. Infelizmente, em geral, não é possível encontrar publicamente softwares que se utilizem de diversas dessas idéias.

Na seqüência apresentamos a idéia de (SHERALI; TUNCBILEK, 1992). Ela é globalmente convergente e encontra o ponto de ótimo global após um número finito de passos (SHERALI; ADAMS, 1999).

4.3 O algoritmo de reformulação–linearização

O algoritmo de *branch-and-bound* descrito em (SHERALI; TUNCBILEK, 1992) é baseado em uma técnica de reformulação-linearização (RL). O foco principal do algoritmo RL é produzir aproximações poliédricas externas ou relaxações baseadas em programação linear próximas ao problema multilinear. Dessa forma, o objetivo é produzir limites superiores para a solução ótima. A chave para obter soluções globais ótimas é embutir essa construção em um esquema de *branch-and-bound*, coordenado com uma estratégia de particionamento que faz com que a diferença entre o limitante superior e a solução ótima diminua a cada passo. Esse esquema resolve o problema multilinear com uma quantidade finita de passos.

Assim, para resolver o problema $MP(\Omega)$, apresentamos um algoritmo *branch-and-bound* que utiliza problemas lineares especialmente construídos usando a técnica de reformulação-linearização (RL). Nesta idéia, geramos restrições multilineares (implícitas no problema

original) tomando produtos de termos em Ω até uma certa ordem. O resultado é então linearizado com a definição de novas variáveis auxiliares, uma para cada termo multilinear que aparece no problema. Esse mecanismo automaticamente cria aproximações externas do fecho convexo da região viável. A idéia principal é substituir cada produto de variáveis $\prod_{j \in J_{rt}} x_j$ por uma nova variável artificial $X_{J_{rt}}$, para todos os termos $t \in T$, obtendo assim um sub-problema linear aproximado (chamado aqui de LP). A solução deste sub-problema é um limite superior para a solução do problema MP correspondente, pois o problema linear LP é uma relaxação do problema MP.

Apenas esta substituição não nos leva a um método com convergência ao ótimo global. Assim, algumas restrições adicionais devem ser incorporadas nos problemas LP para atingir esse objetivo. Para que o método tenha convergência global, iteramos sobre as variáveis, escolhendo para quebrar seus intervalos até que $X_{J_{rt}}$ esteja perto o suficiente de $\prod_{j \in J_{rt}} x_j$. Resolvemos sub-problemas LP sobre cada quebra feita.

Para garantir a convergência, adiciona-se também algumas funções artificiais nos sub-problemas. Chamamos estas funções de artificiais porque elas são redundantes no problema original, mas são relevantes nas versões linearizadas. Estas novas funções artificiais são criadas através de produtos entre as restrições originais do problema e/ou fatores $(x_j - l_j)$ e $(u_j - x_j)$ (chamados de fatores de limite ou *bound-factors* por Sherali e Tuncbilek). Essas funções artificiais são criadas sempre com o cuidado de não aumentar o grau $\delta = \max_{r,t} |J_{rt}|$ do problema original, pois isso pode acarretar em um aumento da complexidade do mesmo (SHERALI; TUNCBILEK, 1992). Como é possível escrever cada restrição original na forma $\phi_r - \beta_r \geq 0$ (se necessário, dividimos equações em duas inequações e multiplicamos inequações inversas por -1) e sabemos que $(x_j - l_j) \geq 0$ e $(u_j - x_j) \geq 0$, multiplicar combinações destas funções sempre cria funções redundantes no mesmo formato. Para ilustrar a idéia, suponha que temos o termo multilinear $x_1 x_2$ no problema original. Criamos as funções artificiais (onde $X_{12} = x_1 x_2$ é uma variável artificial):

$$\begin{aligned} (x_1 - l_1)(x_2 - l_2) &= +X_{12} & -l_2 x_1 & & -l_1 x_2 & & +l_1 l_2 & \geq 0 \\ (x_1 - l_1)(u_2 - x_2) &= -X_{12} & +u_2 x_1 & & +l_1 x_2 & & -l_1 u_2 & \geq 0 \\ (u_1 - x_1)(x_2 - l_2) &= -X_{12} & +l_2 x_1 & & +u_1 x_2 & & -u_1 l_2 & \geq 0 \\ (u_1 - x_1)(u_2 - x_2) &= +X_{12} & -u_2 x_1 & & -u_1 x_2 & & +u_1 u_2 & \geq 0 \end{aligned}$$

Embora sejam redundantes no problema original (pois os limites $l_j \leq x_j \leq u_j$ já estão garantidos no problema MP), na versão linearizada LP estas funções servem como barreiras

para a variável X_{12} (que *a priori* não tem limites por não existir em MP).

Mais formalmente, dado Ω , iniciamos a construção do programa linear $LP(\Omega)$ gerando restrições usando os produtos chamados de *bound-factors*: $(x_j - l_j) \geq 0$, $(u_j - x_j) \geq 0$, $j \in N$, tomando δ fatores por vez. Essas restrições ficam com a forma:

$$F_\delta(J_1, J_2) = \prod_{j \in J_1} (x_j - l_j) \prod_{j \in J_2} (u_j - x_j) \geq 0 \quad (4.1)$$

onde $J_1 \cup J_2 \subseteq N$, $J_1 \cap J_2 = \emptyset$ e $|J_1 \cup J_2| = \delta$.

Após incluir as restrições definidas em (4.1) (chamada de fase de reformulação), passamos para a fase de linearização do problema. Nesta fase, utilizamos a substituição definida por

$$X_J = \prod_{j \in J} x_j, \quad \forall J \in \bigcup_{r=0, \dots, n} \bigcup_{t \in T_r} J_{rt} \quad (4.2)$$

Assumimos que $X_{\{j\}} = x_j$ e $X_\emptyset = 1$. Note que fazemos essa substituição apenas para os J que aparecem como termos do problema $MP(\Omega)$ aumentado.

Podemos afirmar que $v[LP(\Omega)] \geq v[MP(\Omega)]$, onde $v[\cdot]$ denota o valor de ótimo do problema em questão. Além disso, se a solução (x^*, X^*) obtida para o problema $LP(\Omega)$ satisfaz a Equação (4.2), então x^* é uma solução do problema $MP(\Omega)$.

A aproximação dada pela versão linearizada melhora conforme aumentamos o número de funções artificiais, porém devemos limitar essa quantidade — caso contrário os subproblemas lineares ficarão muito grandes e impossíveis de serem resolvidos. Para garantir a convergência desejada é suficiente incluir funções artificiais criadas a partir de produtos de fatores de limite, sem usar nenhuma restrição original do problema como parte de função artificial.

Este método esquematizado aqui é lento quando δ é grande, pois o número de funções artificiais necessárias para garantir a convergência é exponencial em δ . (SHERALI; TUNCBILEK, 1992) mostra que $\sum_{k=0}^{\delta} \binom{m+k-1}{k} \binom{m+(\delta-k)-1}{\delta-k}$ funções artificiais, onde m é o número de variáveis do problema MP, são suficientes para garantir convergência global (note que o número de variáveis no problema MP para resolver uma inferência em rede credal é muito maior que o número de nós da rede; veja as Tabelas 4 e 5 da Seção 4.7). O método de Sherali e Tuncbilek gera todos os produtos de fatores de limite para todas as combinações de variáveis até atingir termos de grau δ . Felizmente o grau máximo dos

problemas MP gerados a partir de uma inferência em uma rede credal não é grande, pois adotamos um processo de transformação da inferência em um problema MP que se utiliza de informações da rede (Seção 4.4.1). Além disso, é possível criar as funções artificiais apenas quando os novos termos que aparecem nelas já são termos do problema multilinear original. Com isso podemos também incluir funções artificiais formadas por fatores de limite e restrições originais sem grande perda de eficiência nos sub-problemas e atingir melhores aproximações.

A execução do algoritmo procede cortando os intervalos das variáveis, sempre limitando esses cortes de acordo com os melhores resultados já obtidos. Assim, vamos introduzir o problema $LP(\Omega)$ em um algoritmo *branch-and-bound* para resolver $MP(\Omega)$. O procedimento envolve o particionamento do conjunto Ω em sub-hiper-retângulos, cada um associado a um nó da árvore de enumeração. Seja $\Omega' \subseteq \Omega$ uma destas partições. Então $LP(\Omega')$ calcula um limite superior para o subproblema $MP(\Omega')$. Em particular, se (\bar{x}, \bar{X}) resolve $LP(\Omega')$ e satisfaz a Equação 4.2, então \bar{x} resolve o problema $MP(\Omega')$. Como neste caso \bar{x} é viável para $MP(\Omega')$, o valor $v[MP(\Omega')] = v[LP(\Omega')]$ é um limitante inferior para o problema $MP(\Omega)$. Temos assim um candidato para atualização da solução global x^* e seu valor v^* . Além disso, se $v[MP(\Omega')] \leq v^*$ então podemos descartar esse nó associado a Ω' , pois nenhuma solução proveniente desse ramo será melhor que a solução já encontrada até o momento.

A cada estágio s do algoritmo, temos um conjunto de nós *ativos* induzido por $q \in Q_s$, cada um associado com uma partição $\Omega^q \subseteq \Omega$. Para cada um destes nós, calculamos o valor B_q pela solução do problema linear $LP(\Omega^q)$. Como resultado temos que o limitante superior para o problema $MP(\Omega)$, em um estágio s do processamento, é dado por

$$B(s) = \min\{B_q : q \in Q_s\}$$

Sempre que a solução do subproblema de um nó torna-se viável no problema $MP(\Omega)$, verificamos se é necessário atualizar a melhor solução atual x^* e seu valor v^* . Além disso, se $B_q \leq v^*$, descartamos o nó q . A lista de nós ativos satisfaz $B_q > v^* \forall q \in Q_s$, em cada estágio s . Selecionamos um nó ativo $q(s)$ que garanta o maior limitante superior dentre os nós $q \in Q_s$, isto é, o nó para o qual $B_{q(s)} = B(s)$, e procedemos decompondo o hiper-retângulo correspondente $\Omega^{q(s)}$ em dois hiper-retângulos, baseado na variável de particionamento x_p selecionada pela regra:

Regra de particionamento:

$$\begin{aligned} p &\in \arg \max_{j \in N} \{\theta_j\} \\ \theta_j &= \max_{t=1, \dots, \delta-1} \max_{J \subseteq N, |J|=t} \{|\hat{X}_{(J \cup \{j\})} - \hat{x}_j \hat{X}_J|\} \text{ para cada } j \in N \end{aligned} \quad (4.3)$$

onde (\hat{x}, \hat{X}) denota a solução ótima obtida para o subproblema do nó correspondente.

Note que a Regra 4.3 de particionamento é baseada em identificar a variável que leva a maior discrepância entre a variável auxiliar e seu produto multilinear associado. A idéia é fazer com que essas diferenças tendam a zero. Cortes nos intervalos das variáveis podem apenas piorar a solução já obtida até aquele momento, mas aproximam a solução da região viável do problema MP. Apresentamos a seguir o algoritmo.

1. **Inicialização.** Inicialize a melhor solução encontrada até o momento com $x^* = \emptyset$ e $v^* = -\infty$ (se um solução viável é conhecida, ela pode ser utilizada aqui). Atribua $s = 1$, $Q_s = \{1\}$, $q(s) = 1$, e $\Omega^1 = \Omega$. Resolva o problema $LP(\Omega)$, obtendo a solução (\hat{x}, \hat{X}) e o valor $B_1 = v[LP(\Omega)]$. Se \hat{x} é viável em $MP(\Omega)$, atualize (se necessário) x^* e v^* . Se $B_1 = v^*$, então x^* resolve $MP(\Omega)$. Caso contrário determine uma variável x_p usando a Regra 4.3.
2. **Particionamento.** Particione o hiper-retângulo selecionado $\Omega^{q(s)}$ em dois sub-hiper-retângulos através da quebra do intervalo de x_p no valor \hat{x}_p .
3. **Calculando limites.** Resolva o problema relaxado para cada um dos dois nós gerados. Atualize (se necessário) a melhor solução atual, e determine uma variável de particionamento, usando a Regra 4.3, para cada um destes dois nós.
4. **Descarte.** Descarte todos os nós que não podem melhorar a solução atual. Isso é feito por

$$Q_{s+1} = Q_s - \{q \in Q_s : B_q + \varepsilon \leq v^*\}$$

onde ε é uma constante medindo a tolerância de otimalidade. Se Q_{s+1} é vazio, termine. Caso contrário, incremente s .

5. **Seleção do nó.** Selecione um nó ativo $q(s) \in \arg \max\{B_q : q \in Q_s\}$. Volte ao passo 2.

A cada corte do algoritmo é possível utilizar algum procedimento de aproximação interna para melhorar a solução atual. Em nossa implementação aplicamos o algoritmo ILS

apresentado na Seção 4.6.3. A melhor solução já encontrada impõe um limite aos cortes; se a solução de um LP de um nó (que serve como aproximação externa) é pior que a melhor solução interna já encontrada, então o nó é descartado, pois nenhuma solução deste nó irá melhorar nossa solução. Note que como possuímos uma solução viável inicial, podemos parar o algoritmo em qualquer ponto, sempre obtendo um intervalo que contém o ponto de ótimo — assim este algoritmo pode ser usado para obtermos uma aproximação (dada por um intervalo) para a solução ótima.

4.4 Algoritmos para inferências em redes credais

Uma *inferência marginal* em uma rede credal é o cálculo de probabilidades superiores e inferiores para um evento na extensão da rede. Digamos que desejamos calcular limites estreitos para $p(q|\mathbb{E})$, onde q é a consulta e \mathbb{E} representa a evidência. É conhecido que a distribuição de probabilidade que minimiza/maximiza $p(q|\mathbb{E})$ está contida no conjunto de vértices da extensão forte da rede (FAGIUOLI; ZAFFALON, 1998), o que torna possível, por exemplo, a construção de algoritmos enumerativos.

$$p(q|\mathbb{E}) = \frac{p(q, \mathbb{E})}{p(\mathbb{E})} = \frac{\sum_{X_i \in \mathbf{X} \setminus (\mathbf{x}_{\mathbb{E}} \cup \{q\})} \prod_{X_i \in \mathbf{X}} p(X_i | \text{pa}(X_i))}{\sum_{X_i \in \mathbf{X} \setminus \mathbf{x}_{\mathbb{E}}} \prod_{X_i \in \mathbf{X}} p(X_i | \text{pa}(X_i))} \quad (4.4)$$

Assim, uma inferência pode ser produzida por otimização combinatória, já que precisamos encontrar um vértice em cada conjunto credal local da rede $K(X_i | \text{pa}(X_i))$ tal que a Expressão (4.4) leva a um máximo/mínimo de $p(q|\mathbb{E})$. Essa inferência cria um grande desafio computacional, sendo normalmente impossível resolver o problema simplesmente fazendo a combinação de todos os possíveis vértices.

O único algoritmo polinomial conhecido é o método 2U, que funciona apenas para redes binárias com topologia em árvore (FAGIUOLI; ZAFFALON, 1998). Outros algoritmos exatos para inferência em redes credais são baseados na enumeração de todos os vértices candidatos na extensão forte que podem produzir valores mínimos/máximos (CANO; CANO; MORAL, 1994; CANO; DELGADO; MORAL, 1993; COZMAN, 2000a; ROCHA; COZMAN, 2002). Assim, poderíamos tentar enumerar todos os vértices possíveis do conjunto credal $K(q|\mathbb{E})$ *a posteriori*. O algoritmo de (CANO; CANO; MORAL, 1994) é um exemplo de algoritmo enumerativo, baseado na transformação (chamada de *CCM*) da rede credal localmente definida em uma rede Bayesiana comum. Sua descrição é simples: suponha

que os conjuntos credais locais $K(X_i|\text{pa}(X_i))$ podem ser combinados de m maneiras; cada combinação é uma função $p_l(X_i|\text{pa}(X_i))$ para $1 \leq l \leq m$. A transformação *CCM* adiciona uma nova variável X'_i para a rede, onde X'_i tem X_i como seu único filho e assume m valores inteiros (ou categorias). Então a variável X_i é relacionada com uma família de densidades de probabilidade condicionais

$$p\left(X_i|\text{pa}(X_i), X'_i = l\right) = p_l(X_i|\text{pa}(X_i)).$$

Adiciona-se uma variável X'_i (chamada de variável transparente) para cada variável X_i da rede credal. Como se sabe que os valores extremos de $p(q|\mathbb{E})$ são atingido nos vértices dos conjuntos credais locais, uma inferência tradicional sobre esta rede Bayesiana produz o resultado desejado. Além disto, cada variável transparente terá tantas categorias quanto existirem vértices no conjunto associado. Isto torna essa transformação inviável para problemas de grande porte, já que a transformação não muda o caráter enumerativo da idéia.

Para ter uma idéia de como as instâncias crescem rapidamente em relação aos parâmetros da rede, pegamos uma rede com três nós, $X \rightarrow Y \leftarrow Z$, onde X , Y e Z possuem quatro categorias cada e todos os conjuntos credais da rede tem somente quatro vértices cada (exemplo retirado de (ROCHA; COZMAN; CAMPOS, 2003)). Como Y tem 16 conjuntos credais, X tem um e Z tem um, existem 4^{16+1+1} distribuições conjuntas diferentes fatorando de acordo com a Expressão (2.1). (ROCHA; COZMAN; CAMPOS, 2003) discute técnicas de *branch-and-bound* que podem tratar situações como esta (porém essas técnicas são facilmente aplicadas apenas para redes com topologia em árvore).

Um caminho diferente para a inferência é perceber que o valor máximo/mínimo para $p(q|\mathbb{E})$ pode ser obtido pela minimização/maximização de uma fração de polinômios multilineares onde técnicas de otimização contínua podem ser aplicadas.

4.4.1 Transformando inferências em problemas multilineares

Uma inferência marginal para a extensão forte pode ser formulada como um problema de programação multilinear. O objetivo é minimizar/maximizar a expressão

$$p(q) = \sum_{X_i \in \mathbb{X} \setminus \{Q\}} \prod_{X_i \in \mathbb{X}} p(X_i | \text{pa}(X_i)) \quad (4.5)$$

sujeito a restrições nas probabilidades locais $p(X_i | \text{pa}(X_i))$ (note que não incluímos a evidência; isso será tratado a seguir). Formulando desta maneira temos que tratar um número enorme de termos não lineares na função objetivo (de fato esse número de termos é exponencial no tamanho da rede). Nesta seção apresentamos uma maneira de reformular a Expressão (4.5), transformando-a em uma coleção de igualdades mais simples.

É possível diminuir o grau máximo dos termos de qualquer problema multilinear com a inclusão de novas variáveis (grau de um termo é a soma dos expoentes de seus fatores), substituindo produtos de variáveis por novas variáveis. Por exemplo o produto xyz pode ser substituído por wz , com a nova restrição $w = xy$ (o termo de grau três foi substituído por dois termos de grau dois). Porém não é garantido que esse tipo de substituição melhore a eficiência dos algoritmos na resolução do problema original, pois aumentamos o número de restrições e o número de variáveis.

Uma transformação eficiente para redes credais, baseada em suas propriedades, é apresentada aqui. Normalmente as transformações que se utilizam de informações conhecidas sobre a estrutura do problema original levam a problemas mais simples de serem resolvidos.

Estamos tratando de inferências sem evidências, porém o caso com evidências pode também ser resolvido utilizando programação multilinear: minimizar $p(q|\mathbb{E})$ é o mesmo que

$$\begin{aligned} & \min t \\ & \text{sujeito a} \\ & p(q, \mathbb{E}) - t \cdot p(\mathbb{E}) \leq 0 \\ & t \geq 0 \end{aligned}$$

Assim, dedicamos o resto da seção para o problema sem evidências, apesar disto não trazer novas complicações.

Para resumidamente ilustrar a idéia da transformação que será proposta, tome uma rede credal em forma de uma cadeia de variáveis $A \rightarrow B \rightarrow C \rightarrow D \rightarrow E$, onde cada variável $X \in \{A, B, C, D, E\}$ é ternária assumindo os valores x_0, x_1 e x_2 . Suponha que queremos encontrar o valor máximo da probabilidade da consulta $E = e_0$; isso pode ser obtido resolvendo

$$\bar{P}(e_0) = \max_{h,i,j,k \in \{0,1,2\}} \sum_{h,i,j,k \in \{0,1,2\}} P(e_0|d_h) P(d_h|c_i) P(c_i|b_j) P(b_j|a_k) P(a_k)$$

sujeito a restrições lineares dos conjuntos credais locais. É necessário portanto lidar com uma função multilinear contendo 81 termos não lineares de grau cinco. No lugar dessa função, podemos transformá-la em um problema multilinear com funções de grau máximo dois, agrupando alguns termos e introduzindo algumas novas variáveis. O resultado é um problema com apenas 30 termos não lineares:

$$\begin{aligned} & \max \sum_{i \in \{0,1,2\}} P(e_0|d_i) P(d_i) \\ & \text{sujeito a} \\ & P(d_i) = \sum_{j \in \{0,1,2\}} P(d_i|c_j) P(c_j), \text{ for } i = 0, 1, 2 \\ & P(c_i) = \sum_{j \in \{0,1,2\}} P(c_i|b_j) P(b_j), \text{ for } i = 0, 1, 2 \\ & P(b_i) = \sum_{j \in \{0,1,2\}} P(b_i|a_j) P(a_j), \text{ for } i = 0, 1, 2 \end{aligned}$$

além das restrições lineares. Essa transformação leva-nos a menos termos não lineares que a versão dada diretamente pela Expressão (2.1). Note também que os termos do problema transformado são de grau menor que no problema original — isto é importante para alguns algoritmos de programação multilinear, como o apresentado na Seção 4.2. Normalmente é mais custoso tratar problemas com graus maiores.

A introdução destas novas funções diminui a quantidade de termos não lineares, diminui o grau máximo do problema, mas aumenta a quantidade de funções não lineares. Assim, devemos tomar o cuidado de escolher os melhores lugares para fazer essa transformação. Em uma rede credal, os lugares ideais para a “quebra” são exatamente nas mensagens transmitidas entre os nós da árvore de eliminação da inferência (COZMAN, 2000b) (como em uma inferência tradicional de uma rede Bayesiana). Assim, o algoritmo de transformação procede de forma semelhante a uma inferência tradicional na rede, es-

colhendo inicialmente uma ordenação para a eliminação das variáveis e então processando tal eliminação. Conforme as variáveis são eliminadas, funções são inseridas ao problema multilinear, criando as relações devidas. No exemplo anterior, a ordem de eliminação de variáveis que produziu o resultado apresentado foi A, B, C, D (E não é eliminada pois é a variável de consulta).

Apresentamos agora o algoritmo para transformar uma inferência sobre uma rede credal em um problema multilinear.

Algoritmo de transformação:

1. Selecione cada conjunto credal local separadamente especificado na rede e crie as funções lineares que o definem (se esses conjuntos credais estão definidos por vértices, então é necessário transformá-los em inequações lineares. Uma maneira de fazê-lo é apresentada em (AVIS, 2000)). Adicione todas estas funções ao problema multilinear. O número de funções inseridas no problema neste passo depende do número de vértices dos conjuntos credais locais e dos tipos de variáveis de probabilidade (binárias, ternárias, etc).
2. Construa uma ordenação para as N variáveis (nós) da rede. A variável de consulta deve ser deixada no final desta ordem. (Essa ordenação pode ser construída de diversas maneiras (COZMAN, 2000b; SHAFER, 1996).) Use essa ordenação para numerar as variáveis, obtendo a seqüência X_N, X_{N-1}, \dots, X_1 . Indicamos a variável de consulta como X_0 . Seja V o conjunto de variáveis X_i . Denote por p_{ij} a probabilidade do evento x_{ij} , isto é, $P(X_i = x_{ij})$.
3. Coloque todas as funções de densidade de probabilidade representadas na rede em um repositório. Estas são as funções que definem relações entre os nós da rede e podem ser escritas como $f(A|B)$, onde $A \subseteq V$, $B \subseteq V$ e $A \cap B = \emptyset$ (note que, inicialmente, $|A| = 1$ e B é o conjunto de pais de A . Assim temos exatamente uma função f para cada nó da rede).
4. Para i de N até 0:
 - (a) Crie uma estrutura de dados, chamada de *Bucket_i*, contendo todas as funções de densidade (do repositório) relacionadas com X_i , isto é, selecione toda

função $f(A|B)$ tal que $X_i \in A \cup B$. Nomeie essas funções escolhidas de $f_1(A_1|B_1), f_2(A_2|B_2), \dots, f_n(A_n|B_n)$.

(b) Para cada função $f_q(A_q|B_q)$ no *Bucket_i*, com q variando de 1 até n , faça

- Suponha que $f_{a_{qu}|b_{qv}}$ é a probabilidade de $A_q = a_{qu}$ dado $B_q = b_{qv}$ (note que a_{qu} é o evento conjunto representando uma combinação de todos os eventos das variáveis em A_q e b_{qv} tem o mesmo significado para B_q). Para cada evento b_{qv} de B_q , gere a função

$$\sum_u p_{a_{qu}|b_{qv}} = 1$$

Em palavras, somamos as probabilidades sobre cada combinação de eventos das variáveis do lado condicionado A_q com a mesma (inalterada em cada função criada) combinação de eventos do lado condicionante B_q . Este processo é repetido para cada combinação do lado condicionante (que fica fixo em cada função). Note que $p_{a_{qu}|b_{qv}}$ são de fato as variáveis do programa multilinear que está sendo criado; seus valores não estão fixos; serão inferidos pelo algoritmo de programação multilinear.

Existem $|a_{qu}| \times |b_{qv}|$ funções criadas aqui, onde $|a_{qu}|$ e $|b_{qv}|$ representam o número de eventos em A_q e B_q , respectivamente. Escreva todas essas funções criadas no problema multilinear.

(c) Multiplique todas as funções de densidade do *Bucket_i* e some esse produto sobre a variável X_i , armazenando o resultado no repositório. Algebricamente esta operação é $\sum_{X_i} \prod_{q=1}^n f_q(A_q|B_q)$, a qual irá gerar uma nova função, chamada aqui de $f_0(A_0|B_0)$ por conveniência. Sabemos que $X_i \notin A_0 \cup B_0$, já que X_i foi somado (eliminado) no processo. A representação detalhada desse processo para o problema multilinear é como segue:

- Suponha que $p_{a_{0u}|b_{0v}}$ é a probabilidade de $A_0 = a_{0u}$ dado $B_0 = b_{0v}$ (note que a_{0u} é o evento representando a combinação de todos os eventos das variáveis em A_0 e mesmo para b_{0v} com relação a B_0). Para cada evento a_{0u} e cada evento b_{0v} , crie a função

$$\sum_{X_i} \left(\prod_{q=1}^n p_{a_{qu}|b_{qv}} \right) = p_{a_{0u}|b_{0v}}$$

onde $p_{a_{qu}|b_{qv}}$ representa a probabilidade do evento $a_{qu}|b_{qv}$ sobre a função $f_q(A_q|B_q)$. (Note que enquanto escrevemos essa função no problema

multilinear, faz-se necessário verificar se a combinação $A_q|B_q$, para todo $q \in \{0, \dots, n\}$, já havia aparecido previamente. Em caso positivo, devemos lembrar de usar os mesmos nomes de variável: caso contrário um nome livre qualquer pode ser escolhido.)

Exatamente $|a_{0u}| \times |b_{0v}|$ funções são criadas aqui, todas elas devem ser acrescentadas ao problema multilinear. Estas funções mostram a relação entre variáveis da rede que aparecem quando fazemos a eliminação de X_i com as variáveis da rede anteriores (antes de multiplicar todas as funções densidade e somar sobre X_i). Note que o processamento do $Bucket_0$ é um pouco diferente, já que não desejamos eliminar a variável de consulta X_0 .

5. Agora é necessário definir somente qual probabilidade p_{0j} de X_0 desejamos minimizar/maximizar.

Obtemos assim um problema de otimização com objetivo e restrições multilineares e lineares, analisando cada nó da árvore de *buckets* gerada durante a eliminação de variáveis (COZMAN, 2000b). Cada uma das restrições lineares e multilineares representa informações locais da rede credal; isto é, as restrições representam os conjuntos credais locais ou relações entre nós vizinhos da árvore de eliminação. O número de funções no problema multilinear gerado é proporcional à quantidade de parâmetros na rede. Esta transformação normalmente pode ser executada de forma eficiente, já que sua complexidade é proporcional a uma inferência em uma rede Bayesiana.

Imagine que um algoritmo de eliminação de variáveis foi executado na rede Bayesiana que tem a mesma estrutura que a rede credal de interesse (isso também vale para SQPNs, por exemplo, já que o caráter da função objetivo é o mesmo). Conforme a árvore de *buckets* é gerada, construímos um problema de otimização com restrições lineares e multilineares para cada *bucket* (COZMAN, 2000b). Assim, o algoritmo segue cada passo do algoritmo de eliminação de variáveis, mas no lugar de fazer os cálculos numéricos, armazena-os como restrições simbólicas. Cada uma dessas restrições representa informações locais da rede, isto é, representam relações entre os *buckets* e variáveis vizinhas.

Para enfatizar a idéia da transformação também para SQPNs, tome uma rede simples com topologia indicada na Figura 12. Todas as variáveis $X \in \{A, B, C, D, E, F\}$ assumem valores $\{x_0, x_1\}$. Suponha que desejamos calcular a influência da variável $F = f_0$ na variável D ; ou seja, precisamos calcular o sinal de $p(d_0|f_0) - p(d_0)$. Isso pode ser feito

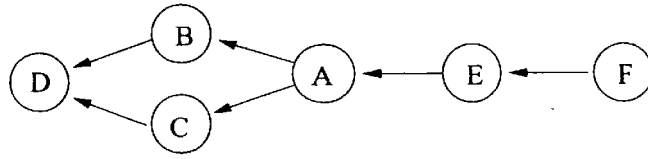


Figura 12: Rede com topologia em losango.

resolvendo a versão “longa” do problema:

$$\begin{aligned}
 & \min / \max \sum_{h,i,j,k} p(d_0|b_i c_j) p(b_i|a_k) p(c_j|a_k) \times \\
 & \times p(a_k|e_h) p(e_h|f_0) - \sum_{g,h,i,j,k} p(d_0|b_i c_j) p(b_i|a_k) \times \\
 & \times p(c_j|a_k) p(a_k|e_h) p(e_h|f_g) p(f_g) \tag{4.6}
 \end{aligned}$$

sujeito a restrições quantitativas e qualitativas.

Temos uma função objetivo multilinear com 48 termos não-lineares de grau cinco e seis. No lugar de lidar diretamente com essa função, podemos utilizar o algoritmo de transformação para gerar funções multilineares mais simples (de grau máximo dois), agrupando termos e introduzindo novas variáveis.

Note que $p(d_0|f_0) - p(d_0)$ é igual a $p(d_0|f_0) p(f_1) - p(d_0|f_1) p(f_1)$ e como estamos interessados no sinal da fórmula, podemos dividi-la por $p(f_1)$ sem alteração no resultado (sinal).

Obtemos um problema multilinear com 36 termos não-lineares no lugar dos 48 necessários antes:

$$\min / \max p(d_0|f_0) - p(d_0|f_1)$$

sujeito a

$$\begin{aligned}
 p(d_0|f_k) &= \sum_j p(d_0|b_j, f_k) p(b_j|f_k), \text{ for } k = 0, 1 \\
 p(d_0|b_i, f_k) &= \sum_j p(d_0|b_i, c_j) p(c_j|f_k), \text{ for } i, k = 0, 1 \\
 p(b_i|f_k) &= \sum_j p(b_i|a_j) p(a_j|f_k), \text{ for } i, k = 0, 1 \\
 p(c_i|f_k) &= \sum_j p(c_i|a_j) p(a_j|f_k), \text{ for } i, k = 0, 1 \\
 p(a_i|f_k) &= \sum_j p(a_i|e_j) p(e_j|f_k), \text{ for } i, k = 0, 1
 \end{aligned}$$

além das restrições quantitativas e qualitativas. Essa redução no número de termos não-lineares é exponencialmente mais efetiva com redes de tamanho maior.

(ANDERSEN; HOOKER, 1994) trata o problema da inferência sobre redes credais de forma similar, também seguindo a linha da transformação para um problema de otimização matemática. Porém em seu trabalho a transformação é executada de maneira diferente (eles utilizam uma descrição estendida de mundos possíveis ou *possible worlds*, o que leva à extração de outras funções para representar as restrições de independência da rede), levando a sub-problemas não lineares complicados. A vantagem do método apresentado aqui está em sua formulação direta, no sentido que a geração do problema multilinear segue exatamente os mesmos passos que uma inferência tradicional sobre uma rede Bayesiana. Devido a isso, podemos facilmente executar algoritmos de aproximação especializados no problema credal (como o algoritmo A/R++ descrito na Seção 4.6.2), obtendo intervalos mais próximos para as variáveis e conseqüentemente um otimizador multilinear mais eficiente.

As idéias apresentadas nesta seção mostram uma maneira de resolver inferências em redes credais utilizando programação multilinear. Na Seção 4.6 voltamos a esse tema, mas com o intuito de apresentar idéias que podem ser utilizadas para a construção de algoritmos aproximados.

4.5 Inferências em redes semi-qualitativas

Restringindo o foco para as redes qualitativas (QPNs), uma inferência refere-se a questão qualitativa de como a observação de algumas variáveis altera a probabilidade de outras. Se $Q = q$ é a consulta e \mathbb{E} a evidência, precisamos calcular $P(q|\mathbb{E}) - P(q)$. Quando $\max(P(q|\mathbb{E}) - P(q)) \leq 0$, temos uma influência negativa da observação sobre Q . Se $\min(P(q|\mathbb{E}) - P(q)) \geq 0$, então temos uma influência positiva. Se ambos \max e \min são zero, então não temos influência. Caso contrário temos uma influência ambígua da observação na variável de consulta. Um algoritmo polinomial para inferências em QPNs foi proposto em (DRUZDZEL; HENRION, 1993): analisando efeitos, o algoritmo gera um sinal para cada nó, indicando a direção de mudança na probabilidade do nó, de acordo com a nova observação (dados todos os valores prévios). Diversos outros algoritmos eficientes para inferências em QPNs já foram propostos, incluindo algoritmos que tratam múltiplas observações (RENOOIJ; GAAG; PARSONS, 2000), sinais ambíguos (BOLT; RENOOIJ; GAAG, 2003), influências não monotônicas (RENOOIJ; GAAG, 2000) e outras relações (BOLT; GAAG; RENOOIJ, 2004, 2003; RENOOIJ; GAAG, 1999; RENOOIJ; GAAG; PARSONS, 2002; GAAG; BODLAENDER; FEELDERS, 2004).

Para SQPNs, o cálculo de inferências *exatas* é geralmente um desafio mais complexo, e os algoritmos existentes focam em soluções aproximadas (RENOOIJ; GAAG, 2002). Além de lidar com as relações qualitativas, é necessário tratar as restrições numéricas. De fato esse problema pode ser tão difícil quanto as inferências em redes credais, como mostramos no Capítulo 3.

Uma inferência em SQPN pode ser formulada com um problema de programação não-linear. O objetivo é minimizar/maximizar a expressão $P(q|\mathbb{E}) - P(q)$, sujeito a restrições numéricas e relações qualitativas; dependendo dos sinais produzidos na minimização/maximização, sabemos como e influencia Q . A consulta pode ser escrita usando uma expressão (multilinear) básica das redes Bayesianas:

$$P(q|\mathbb{E}) = \sum_{Y \in \{X_i\}} \prod_{Y \in \{X_i, Q\}} p(Y|\text{pa}(Y), \mathbb{E}), \quad (4.7)$$

onde $X_i \notin \{Q\} \cup \mathbb{E}$. Para calcular $P(q)$ sem evidência, apenas removemos todos os \mathbb{E} da expressão. Como descrito nas seções 2.4.1 e 2.4, todas as restrições qualitativas podem ser escritas de forma linear e multilinear, usando as expressões (2.3), (2.4), (2.5) (2.6), (2.7) e (2.8).

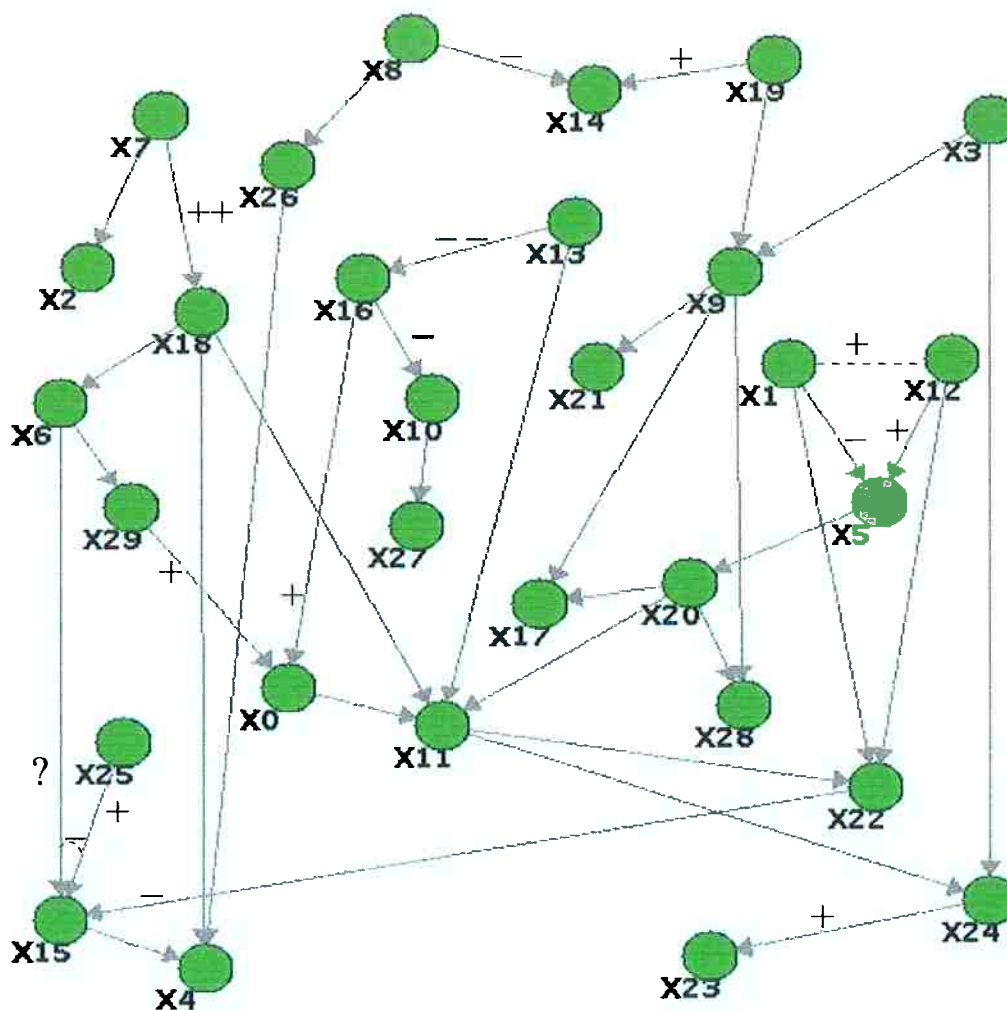


Figura 13: Um exemplo de rede semi-qualitativa.

A dificuldade que aparece é a mesma encontrada nas redes credais: a Expressão (4.7) potencialmente contém uma quantidade enorme de termos multilineares (essa quantidade é exponencial no tamanho da rede). Propomos transformar a Expressão (4.7) em uma coleção de restrições menores, da mesma forma que é feito para as redes credais (Seção 4.4). Outros autores (ANDERSEN; HOOKER, 1994; DRUZDZEL; GAAG, 1995) propõe transformações que produzem programas multilineares para modelos gráficos, porém tais propostas são menos efetivas (usando idéias de *mundos possíveis*) e menos aplicáveis às SQPNs. A idéia proposta aqui tem ainda o benefício de poder ser utilizada para encontrar soluções aproximadas, através das diversas técnicas existentes e aplicáveis em programação multilinear.

Finalmente enfatizamos que a idéia proposta é eficiente na prática, sendo capaz de tra-

tar redes com 150 nós e uma quantidade significativa de relações qualitativas (geradas aleatoriamente (IDE; COZMAN, 2002)).

Para ilustrar o método de programação multilinear, considere a rede SQPN da Figura 13. Nós sem sinais qualitativos estão associados com distribuições de probabilidades. Diversos tipos de relações qualitativas estão presentes na rede, como:

$$\begin{aligned} &S^+(X_{25}, X_{15}), S^-(X_{22}, X_{15}), Y^-(\{X_6, X_{25}\}, X_{15}), \\ &S^{++}(X_7, X_{18}), S_{x_{25}}^{?(+)}(X_6, X_{15}), S_{x_{25}}^{?(-)}(X_6, X_{25}), \\ &S^+(X_{16}, X_0), S^+(X_{29}, X_0), S^-(X_8, X_{14}), S^+(X_{19}, X_{14}), \\ &S^{--}(X_{13}, X_{16}), X^+(\{X_1, X_{12}\}, x_5), S^-(X_1, X_5), \end{aligned}$$

onde S^{++} e S^{--} representam influências fortes. O método multilinear RL encontra prontamente que a observação $\{X_{13} = x_{13}\}$ influencia x_4 negativamente.

4.6 Soluções aproximadas para inferência em redes credais

O método RL proposto na Seção 4.3 é geral para problemas multilineares; aplicado em inferências sobre redes credais, não se beneficia de informações importantes que são conhecidas sobre a rede. Assim, descrevemos nesta seção dois algoritmos aproximados desenvolvidos neste trabalho que se utilizam da estrutura da rede credal e por isso tendem a ser mais eficientes. Além disto citamos como combinar essas idéias aproximadas para melhorar o desempenho do método RL.

Existem basicamente dois tipos de aproximação: interna e externa. A aproximação interna produz intervalos que estão contidos entre o máximo e mínimo corretos, enquanto que a aproximação externa produz intervalos que encapsulam o mínimo e o máximo exatos. O algoritmo A/R++ é um algoritmo de aproximação externa derivado do algoritmo A/R+; outros algoritmos desse tipo podem ser encontrados em (CANO; MORAL, 2002; HA, 1998; TESSEM, 1992). O algoritmo ILS é de aproximação interna; algoritmos de aproximação interna podem ser encontrados em (ANDERSEN; HOOKER, 1994; CANO; CANO; MORAL, 1994; CANO; MORAL, 1996; COZMAN, 1997; ROCHA; COZMAN; CAMPOS, 2003). A grande vantagem de poder executar um algoritmo de aproximação externa e outro de aproximação interna é obter um intervalo (normalmente pequeno) que encapsula o resul-

tado correto, já que cada uma destas aproximações tendem ao resultado correto por um lado diferente. Dessa forma podemos dizer que a junção do algoritmo A/R++ com o ILS leva-nos a produzir aproximações mais precisas para o problema de inferência.

4.6.1 A/R+ para redes multi-conectadas

O desempenho do algoritmo RL apresentado na Seção 4.3 é melhorado se aproximações para os intervalos de algumas ou todas as variáveis são conhecidos (SHERALI; TUNCBILEK, 1992). Uma maneira de obter aproximações para os intervalos das variáveis é usar o algoritmo A/R+ (ROCHA; COZMAN; CAMPOS, 2003). O algoritmo A/R+ (e seu antecessor A/R proposto em (TESSEM, 1992)) produz aproximações locais para as mensagens enviadas entre nós durante a inferência. Um nó envia/recebe intervalos aproximados de/para seus pais e filhos; estas aproximações são rapidamente calculadas e transmitidas. Note que os algoritmos A/R e A/R+ foram desenvolvidos para funcionar apenas em redes com topologia em árvore (o grafo subjacente à rede não possui circuitos).

Para tratar redes multi-conectadas, o algoritmo A/R+ deve ser processado sobre a árvore de eliminação de variáveis, e não sobre a rede original (que neste caso pode ser multi-conectada). Neste caso o algoritmo A/R+ é executado sobre a árvore de eliminação de variáveis de forma análoga ao algoritmo de transformação descrito na Seção 4.4.1. As funções não lineares geradas no passo 4(c) daquele algoritmo representam as probabilidades da variável eliminada. Podemos assim, no momento de executar aquele passo, minimizar e maximizar localmente aquelas funções, obtendo limites em seus valores. Estes limites são as mensagens enviadas entre os nós da rede. A propagação dessas mensagens na rede induz intervalos que contêm as probabilidades corretas para cada nó da rede. Olhando para o nó de consulta, obtemos um intervalo contendo a solução desejada.

Na Figura 14 temos uma rede multi-conectada com seis variáveis. O algoritmo A/R+, aplicado a esta rede multi-conectada e tendo F como variável de consulta, será processado sobre a árvore de eliminação. Uma possível árvore de eliminação das variáveis é dada na Figura 15. A cada passo, sobre essa árvore de eliminação, o algoritmo A/R+ para redes multi-conectadas calcula os valores dados a seguir e propaga-os pela rede.

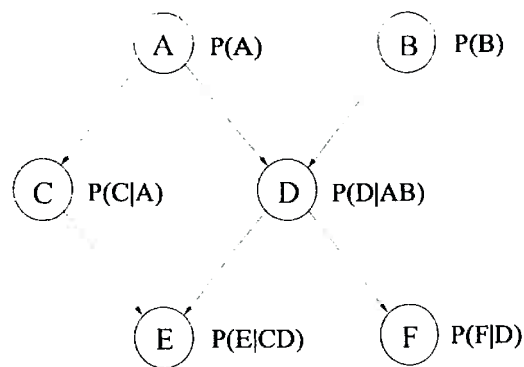


Figura 14: Rede multi-conectada com seis vértices.

$$\min/\max P(e_i|a_j d_k) \quad \forall i, j, k$$

$$\min/\max P(d_i|a_j) \quad \forall i, j$$

$$\min/\max P(d_i e_j) \quad \forall i, j$$

$$\min/\max P(d_i) \quad \forall i$$

$$\min/\max P(f_i) \quad \forall i$$

O fato importante a ressaltar é que mensagens no algoritmo A/R+ (e sua generalização para redes multi-conectadas) são obtidas por procedimentos de otimização local; em fato, estes problemas de otimização são simplesmente problemas de programação multilinear. Assim é possível resolver pequenos problemas multilineares em cada passo do algoritmo A/R+, encolhendo os intervalos sempre que os cortes de variáveis são executados no procedimento de *branch-and-bound* do algoritmo RL. Para resolver esses problemas locais, utilizamos o próprio algoritmo RL.

4.6.2 A/R++

As informações passadas pelo algoritmo A/R+ representam uma coleção de probabilidades inferiores e superiores para os valores de cada variável. Porém, com a técnica de programação multilinear, podemos passar mais informações: é possível executar diversas otimizações locais, obtendo valores inferiores e superiores para diversos eventos, e transmitir toda essa informação estendida da mesma forma que no algoritmo A/R+. Chamamos

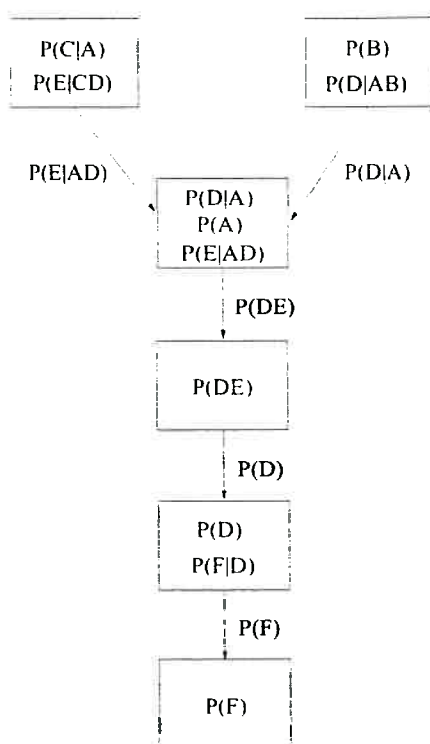


Figura 15: Possível árvore de eliminação das variáveis para a rede da Figura 14.

esse algoritmo resultante de A/R++.

A idéia do algoritmo A/R++ é avaliar não apenas intervalos para os valores das variáveis eliminadas em cada nó, mas também intervalos para outros eventos definidos naquele nó, sempre processando apenas informações locais. Em nossa implementação atual escolhemos calcular $\min / \max \alpha + \beta$, para toda combinação de valores atômicos α e β em cada nó. Em outras palavras, enquanto o algoritmo A/R+ apenas minimiza e maximiza as probabilidades dos eventos relacionados ao nó, o algoritmo A/R++ também minimiza e maximiza a soma de pares de eventos, incluindo assim mais restrições nos valores e gerando uma melhor aproximação. Apesar da escolha de trabalhar com somas de pares de eventos, qualquer outra função linear baseada nas variáveis locais poderia ser processada. Esses novos limites não são redundantes, pois os limites que possuíamos antes daqueles cálculos eram que $\min(\alpha + \beta) \geq \min \alpha + \min \beta$ (e similarmente para a maximização). Note que os intervalos calculados pelo A/R++ são sempre mais precisos (ou iguais) que os intervalos do algoritmo A/R+, já que informações adicionais são transportadas (além das informações do A/R+). Estas novas informações que precisam ser propagadas não trazem nenhuma dificuldade computacional, já que sempre podem ser representadas por funções lineares. Além disso, a representação através de funções é especialmente útil para

o algoritmo RL descrito na Seção 4.2.

Voltando ao exemplo apresentado nas Figuras 14 e 15, o algoritmo A/R++ calcularia as seguintes probabilidades:

$$\begin{array}{lll}
 \text{min/max} & P(e_i|a_j d_k) + P(e_l|a_m d_n) & \forall i, j, k, l, m, n \\
 \text{min/max} & P(d_i|a_j) + P(d_k|a_l) & \forall i, j, k, l \\
 \text{min/max} & P(d_i e_j) + P(d_k e_l) & \forall i, j, k, l \\
 \text{min/max} & P(d_i) + P(d_j) & \forall i, j \\
 \text{min/max} & P(f_i) + P(f_j) & \forall i, j
 \end{array}$$

Obviamente é necessário ter o cuidado de propagar uma quantidade “gerenciável” de informações. Notamos que a propagação proposta (através da soma de pares de eventos) gera uma quantidade quadrática de informações em relação à quantidade transmitida pelo algoritmo A/R+.

Usando o algoritmo A/R++ para calcular intervalos iniciais para o problema MP, obtemos uma melhor aproximação inicial. Além disso, é possível executá-lo para redução dos intervalos a cada passo do método *branch-and-bound*. (SHERALI; ADAMS, 1999) sugere que técnicas de redução dos intervalos das variáveis podem ser utilizadas; implementamos esta técnica usando o A/R++ em cada nó do procedimento de corte dos intervalos (*branch*), o que tende a reduzir o número de cortes totais necessários para atingir o ótimo.

4.6.3 Busca Local Iterativa

Uma aproximação interna para os extremos de $p(q|\mathbb{E})$ pode ser gerada por um método que procura por um máximo local de $p(q|\mathbb{E})$ sujeito às restrições impostas pelos conjuntos credais locais $K(X_i|pa(X_i))$. Tais aproximações já foram consideradas na literatura, usando por exemplo *simulated annealing* (CANO; CANO; MORAL, 1994) e algoritmos genéticos (CANO; MORAL, 1996). Geralmente estes métodos necessitam especificar diversos parâmetros de controle da execução: embora eles produzam resultados razoáveis, em geral são difíceis de serem aplicados. Outras idéias, como busca através de gradientes

ou programação geométrica também tem sido propostas, porém nenhuma implementação é conhecida até o momento (ANDERSEN; HOOKER, 1994; COZMAN, 1997; ZAFFALON, 1997) — tais técnicas podem demandar grande cuidado na especificação dos parâmetros e precisam ser cuidadosamente adaptadas para funcionar com redes credais.

O cálculo dos extremos de $p(q|\mathbb{E})$ é um problema de maximização com restrições em $K(X_i|pa(X_i))$ e uma função objetivo que é uma razão entre duas funções multilíneas grandes, representada por (2.1) (tal problema é usualmente classificado como um programa *signomial* (AVRIEL, 1980)). Nesta seção analisamos o fato que, em nosso problema, 1) todas as restrições são lineares; 2) todas as restrições são locais (elas podem ser agrupadas de acordo com os conjuntos credais locais $K(X_i|pa(X_i))$); e 3) funções multilíneas são funções signomiais relativamente simples.

Apresentamos um novo algoritmo que se beneficia destas propriedades, sem parâmetros livres para serem especificados e sem a necessidade de métodos especiais para controlar problemas numéricos. Esse algoritmo é inspirado no trabalho de Lukatskii e Shapot para problemas multilíneas (LUKATSKII; SHAPOT, 2000). Lukatskii e Shapot tratam o problema de uma função multilinear a ser otimizada, sujeita a restrições lineares. A idéia de seu algoritmo é simples: fixe variáveis no problema multilinear até atingir um problema linear, que pode ser visto como uma aproximação para o problema multilinear. Iteramos esse processo, alternando as variáveis fixadas e convergindo a um ponto de ótimo. Aplicamos o mesmo princípio no nosso problema: fixamos todas as variáveis exceto uma, e resolvemos o problema linear resultante. Fixamos outra variável, resolvendo o novo problema linear, e assim sucessivamente, até que nenhuma melhora no valor de ótimo seja possível localmente. O algoritmo certamente termina, pois o programa multilinear tem seu ponto de máximo/mínimo em um vértice da região viável.

Nossa inferência não é exatamente um programa multilinear, já que a função objetivo é uma fração de funções multilíneas. Porém as inferências mantêm uma propriedade essencial usada pelo algoritmo de Lukatskii-Shapot: todo ponto de máximo/mínimo de $p(q|\mathbb{E})$ em uma extensão forte ocorre em um vértice da extensão (FAGIUOLI; ZAFFALON, 1998). Isto é, podemos fixar um vértice para cada conjunto credal local exceto um deles, e apenas verificar qual vértice do conjunto não fixado minimiza/maximiza $p(q|\mathbb{E})$ (ou seja, um problema linear). Guardamos o vértice minimizador/maximizador e passamos para o próximo conjunto credal local. Fixamos novamente todos os vértices nos conjuntos credais exceto este que estamos tratando, obtendo um novo vértice para ele. Repetimos

esse processo, tratando cada conjunto credal local da rede. Como a cada passo certamente melhoramos o valor da função objetivo, e somente um número finito de alterações pode ser feita (dado que as variáveis são discretas e os conjuntos credais locais tem número finito de vértices), asseguramos que o procedimento terminará em um ótimo local.

Essa busca local descrita pode ser facilmente implementada como segue. Assuma que estamos buscando pela probabilidade máxima (ou superior) de $p(q|\mathbb{E})$.

Algoritmo ILS

1. Escolha uma ordenação para os nós da rede credal.
2. Escolha um vértice para cada conjunto credal local, e calcule $p(q|\mathbb{E})$ usando estes vértices (este é o chute inicial).
3. Repita até que nenhuma melhoria no valor de $p(q|\mathbb{E})$ seja encontrada, percorrendo indefinidamente todos os nós da rede de acordo com a ordenação (o nó corrente é denotado por X_i):
 - (a) Mantenha todos os vértices fixados, exceto no conjunto credal $K(X_i|pa(X_i))$ associado ao nó X_i .
 - (b) Execute uma otimização sobre a rede, verificando qual vértice $v \in K(X_i|pa(X_i))$ maximiza localmente $p(q|\mathbb{E})$; fixe $p(X_i|pa(X_i))$ neste vértice v .
4. Pare quando não existir alteração no valor de ótimo e retorne a $p(q|\mathbb{E})$ atual.

Este algoritmo pode ser trivialmente alterado para aproximar probabilidades inferiores. Perceba também que o algoritmo é completamente geral, não restrito a topologias em árvore, mas é particularmente eficiente nestes casos. Finalmente note que existem diversas idéias para a escolha da ordenação dos nós gerada no primeiro passo; em fato algumas ordenações podem ser melhores que outras, mas a escolha da melhor ordenação em si já é um problema difícil.

Na próxima seção apresentamos resultados empíricos com os algoritmos aproximados apresentados aqui e comparamos com os resultados do algoritmo RL da Seção 4.3.

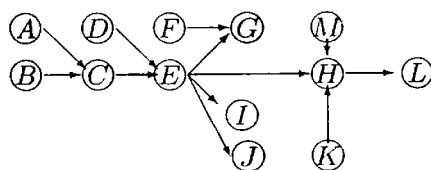


Figura 16: Uma rede credal com 13 nós e topologia em árvore.

4.7 Resultados experimentais

Apresentamos nesta seção resultados experimentais com os algoritmos desenvolvidos.

4.7.1 Busca Local Iterativa

De forma geral, o algoritmo ILS proposto produz resultados muito precisos. Executamos o algoritmo em uma grande quantidade de redes de médio porte e constatamos que na maioria dos casos ele encontra a solução exata (fizemos uso do algoritmo de *branch-and-bound* de (ROCHA; COZMAN; CAMPOS, 2003) e/ou do algoritmo RL para calcular os valores exatos).

Um problema para comparar esse algoritmo aproximado com algoritmos exatos é a falta de procedimentos exatos que possam tratar redes de grande porte. Para termos uma idéia da precisão deste algoritmo, consideramos a rede apresentada na Figura 16. Foram geradas 30 redes credais diferentes sobre o mesmo grafo indicado na figura. As variáveis foram criadas com 2 a 4 categorias e os conjuntos credais de 2 a 3 vértices. Essa rede, apesar de pequena, já gera dificuldades computacionais para a resolução através de algoritmos exatos. Pudemos verificar nos testes que o algoritmo ILS produziu respostas em tempo inferior a três minutos para todos os testes, e acertou exatamente o ponto de ótimo global em mais da metade dos casos. Nos demais, o erro foi inferior a 0.1%.

Os experimentos realizados indicam que a utilização de um método de aproximação interna em conjunto com um método de aproximação externa (como o A/R++) produz um intervalo preciso contendo a solução ótima. Encontrar tal intervalo é uma tarefa extremamente menos custosa que executar um algoritmo exato para a inferência. A Seção 4.7.2 mostra testes realizados com o algoritmo A/R++ (aproximação externa) em conjunto com o algoritmo ILS e o algoritmo RL para a produção de resultados exatos.

Conjunto de testes	Topologia da rede	Número de nós na rede	Número de vértices por conjunto credal local
A	dense binary	10	2
B	binary Alarm	37	2
C	dense ternary	10	3
D	ternary Alarm	37	3
E	dense quaternary	10	4

Tabela 4: Formato dos conjuntos de testes.

4.7.2 RL e A/R++

Para ilustrar o comportamento do método RL, apresentamos dois conjuntos de experimentos. Inicialmente trabalhamos com conjuntos de testes divididos em dois tipos: redes densas com dez variáveis (aleatoriamente geradas usando o software BNGenerator (IDE; COZMAN, 2002)) e redes com topologia igual a da rede *Alarm* (BEINLICH *et al.*, 1989). Depois mostramos alguns exemplos gerais de grande porte.

As Tabelas 4 e 5 mostram resultados de cinco diferentes tipos de redes. Cada linha na Tabela 5 mostra a média dos resultados do processamento de dez redes credais multiconectadas geradas aleatoriamente. Todos os testes feitos são maximizações de probabilidades e foram executados transformando as inferências em problemas de programação multilinear. As variáveis de consulta nas inferências foram escolhidas de forma a criar o maior desafio possível (isto é, escolhemos sempre as variáveis mais difíceis de serem avaliadas da rede). As redes tem diferentes tipos de variáveis (binárias, ternárias e quaternárias) e diferentes tamanhos por conjunto credal local de cada nó da rede.

A Tabela 5 mostra o tamanho do problema multilinear gerado e o tamanho das aproximações linearizadas do mesmo; note que o tamanho do problema multilinear cresce substancialmente com o número de vértices por conjunto credal e com o número de variáveis na rede original. Enfrentamos dificuldades computacionais para tratar os sub-problemas lineares das últimas duas linhas da tabela (na última linha aparecem sub-problemas com mais de trinta mil variáveis e duzentas mil restrições, e estes não puderam ser resolvidos).

Além disto, a Tabela 5 mostra o erro médio induzido pelos intervalos contendo o valor ótimo, gerados pelos algoritmos A/R++ e RL (que internamente também usa o A/R++), e o número médio de nós avaliados no procedimento de *branch-and-bound*.

Conjunto de testes	Variáveis # em MP	Funções em MP	Variáveis linearizadas	Funções linearizadas
A	105	172	665	3996
B	363	576	1395	6876
C	412	576	5920	40181
D	1657	2214	13780	70612
E	1145	1474	30073	213376

Conjunto de testes	Erro do A/R++	Erro do RL	Nós B&B do RL
A	2.8684%	0.0484%	301
B	5.5706%	1.076%	765
C	10.4304%	0.3290%	1
D	22.3293%	2.5954%	3
E	13.4146%	0.6071%	1

Tabela 5: Tamanho médio dos problemas multilineares e suas versões linearizadas correspondentes. Erros induzidos e média de nós avaliados.

Para ilustrar o esforço do algoritmo RL, a última coluna mostra o número de nós avaliados durante o *branch-and-bound*. Note que no conjunto de testes D apenas 80% das inferências nas redes puderam ser resolvidas e no conjunto E menos de 50%. Todos os testes foram feitos em um computador Pentium IV 1.7Ghz com 1GB de RAM, tempo limite de dez minutos por teste nos conjuntos A, B e C, e uma hora para os testes nos conjuntos D e E.

A Tabela 6 mostra as soluções produzidas para casos de teste de oito até cento e vinte e seis variáveis, com topologias geradas aleatoriamente. Até onde sabemos nenhum outro algoritmo existente pode processar redes com estes tamanhos. Todos os testes foram realizados com um tempo limite de cinco minutos cada (nas últimas duas linhas usamos limite de 30 minutos). Como cada solução global pertence ao intervalo apresentado, o erro foi calculado dividindo o valor médio do intervalo pelo seu valor mínimo. Se restringirmos os testes apenas para redes com topologia em árvore, o método RL pode tratar redes maiores com menos erro.

Os resultados produzidos parecem promissores, superando em geral os algoritmos atuais para inferência em redes credais. Mesmo assim notamos no final da Tabela 6 que conforme as redes crescem os algoritmos tornam-se inviáveis (do ponto de vista de obter soluções exatas). Nestes casos é possível executar métodos aproximados, mas não temos acesso aos valores ótimos para uma comparação mais apurada.

Nós na rede	Variáveis multilineares	Funções multilineares	Solução usando RL	Erro
8	53	65	[0.6376, 0.6376]	0.0%
13	223	351	[0.2590, 0.2595]	0.2%
37	2042	777	[0.5605, 0.5645]	0.7%
84	441	781	[0.7803, 0.7811]	0.1%
96	437	772	[0.6794, 0.6885]	1.3%
126	1117	1751	[0.8531, 0.9985]	14.5%
126	8211	9291	[0.9199, 1.0000]	8.0%

Tabela 6: Testes com redes geradas aleatoriamente.

5 *Considerações finais*

Discutimos neste trabalho a complexidade computacional e novos algoritmos para inferência em redes credais e semi-qualitativas.

Podemos resumir os resultados de complexidade como segue. Sobre redes Bayesianas, introduzimos o conceito de rede com largura limitada, demonstrando que muitos problemas onde a largura cresce logaritmicamente com a rede têm soluções polinomiais. Mostramos também a diferença entre os problemas BN-MPE e BN-MPEe. Além disso, introduzimos o problema **MmAP**, descrevendo sua complexidade. Ainda resta um estudo mais detalhado sobre esse problema e suas aplicações.

Com relação às redes credais, apresentamos a complexidade computacional dos problemas CN-MPE, CN-MPEe, e CN-MAP. O caráter polinomial do problema CN-MPE, para alguns casos, é surpreendente. Esses resultados têm implicação em outros modelos que lidam com conjuntos e intervalos de probabilidades, funções de crença, medidas de possibilidades, probabilidades qualitativas e lógica probabilística. Uma futura contribuição seria o estudo mais detalhado da complexidade de CN-MAP para redes multi-conectadas.

Sobre as SQPNs, caracterizamos a complexidade de inferências exatas sobre diversos tipos de redes. Mostramos que, se exigirmos resultados exatos, a complexidade de tempo em pior caso das inferências em SQPNs e nas redes credais é a mesma. Assim, idéias para algoritmos de uma das classes podem surgir a partir das idéias da outra. Como algumas SQPNs são tão difíceis quanto redes credais, precisamos inevitavelmente estar preparados para técnicas sofisticadas, como por exemplo programação multilinear.

Existem diversos problemas interessantes ainda a serem tratados. Problemas sobre redes binárias talvez apresentem menor dificuldade e resultados de complexidade sobre elas podem ser explorados para PT-BN-MAP, BIW-BN-MAP e BIW-CN-Pr. Estes problemas claramente pertencem a NP e estão relacionados entre eles, mas a questão se existem

algoritmos polinomiais para resolvê-los ainda não foi respondida. Notamos apenas que para o problema CN-Pr existem algoritmos polinomiais para árvores binárias (FAGIUOLI; ZAFFALON, 1998).

Com relação aos algoritmos, a principal contribuição é o uso de técnicas de programação multilinear na resolução dos problemas. Embora alguns autores já tenham sugerido este caminho para processar inferências, nenhuma investigação mais profunda ou implementação havia sido conduzida.

Os resultados produzidos através de experimentos parecem promissores em comparação com algoritmos atuais para inferência em redes credais. Outras técnicas para resolver os problemas multilineares podem ser investigadas no futuro, talvez criando uma combinação entre técnicas (como métodos de planos de corte e procedimentos de buscas locais) para atingir melhores resultados e eficiência.

Para redes credais muito grandes, mesmo as técnicas de programação multilinear são insuficientes. Neste caso executar as inferências de forma aproximada parece um caminho natural a seguir. Como já indicado pelo algoritmo A/R++ e pelo algoritmo ILS, a programação multilinear pode ser muito útil para a criação de algoritmos aproximados.

Pretendemos, na continuação da pesquisa, aprofundar os estudos sobre outros algoritmos para programação multilinear, possibilitando assim a resolução de inferências mais complexas. Além disso, algoritmos eficientes para programação multilinear são facilmente aplicados em diversas áreas da Computação, como geometria computacional e otimização combinatória. Pretendemos utilizar esses algoritmos sobre outros problemas para comparar os desempenhos com as técnicas normalmente utilizadas.

Restringindo o foco para as SQPNs, apresentamos como aplicar um algoritmo para utilizar programação multilinear na resolução de inferências. Esse algoritmo é capaz de lidar com todas as relações qualitativas sugeridas na literatura para QPNs e diversas restrições numéricas (precisas e imprecisas).

Dessa forma sugerimos que a abordagem multilinear pode ser vista como um ponto de encontro para diversos formalismos de incerteza, sendo certamente uma maneira viável em problemas práticos. Por exemplo, podemos facilmente misturar restrições lógicas com restrições probabilísticas (NILSSON, 1986). De fato podemos ir mais longe, explorando restrições que simultaneamente incluem elementos lógicos e probabilísticos (CAMPOS;

COZMAN, 2005a). Algoritmos para inferência em lógica probabilística existentes na literatura usualmente reduzem as restrições em problemas lineares, desprezando o caráter não-linear das restrições de independência (GAAG, 1991; HAILPERIN, 1996; HANSEN; JAUMARD, 1996). Na abordagem multilinear, conseguimos representar as restrições de lógica probabilística sem desprezar os julgamentos de independência. Outras extensões podem ser facilmente introduzidas, como inferências sob julgamentos de irrelevância e independência epistêmica (CAMPOS; COZMAN, 2005b).

Deixamos aqui a possibilidade de aprofundar os estudos da combinação de redes credais e SQPNs com dados empíricos para aprendizado. Idéias interessantes, como métodos Bayesianos baseados no modelo impreciso de Dirichlet (IDM) podem ser explorados (CAMPOS; COZMAN, 2005a), levando-nos novamente para a necessidade de programação multilinear e o tratamento de redes credais.

Referências

- ANDERSEN, K. A.; HOOKER, J. N. Bayesian logic. *Decision Support Systems*, v. 11, p. 191–210, 1994.
- AVIS, D. lrs: A revised implementation of the reverse search vertex enumeration algorithm. *Polytopes - Combinatorics and Computation*, Birkhauser-Verlag, p. 177–198, 2000. DMV Seminar Band 29.
- AVRIEL, M. *Advances in Geometric Programming*. New York: Plenum Press, 1980.
- AVRIEL, M.; DEMBO, R.; PASSY, U. Solution of generalized geometric programs. *International Journal for Numerical Methods in Engineering*, v. 9(149), 1975.
- BAILEY, D. D.; DALMAU, V.; KOLAITIS, P. G. Phase transitions of PP-complete satisfiability problems. In: *International Joint Conference on Artificial Intelligence*. Seattle, Washington, 2001. p. 183–192.
- BEINLICH, I.; SUERMONDT, H. J.; CHAVEZ, R. M.; COOPER, G. F. The ALARM monitoring system: A case study with two probabilistic inference techniques for belief networks. In: *II European Conf. on Artificial Intelligence in Medicine*. Berlin: Springer-Verlag, 1989. p. 247–256.
- BOLT, J. H.; GAAG, L. C. van der; RENOOIJ, S. Introducing situational influences in QPNs. In: *ECSQARU*, 2003. p. 113–124.
- BOLT, J. H.; GAAG, L. C. van der; RENOOIJ, S. The practicability of situational signs for QPNs. In: *IPMU*. Perugia, Italy, 2004. p. 1691–1698.
- BOLT, J. H.; RENOOIJ, S.; GAAG, L. C. van der. Upgrading ambiguous signs in QPNs. In: *UAI*, 2003. p. 73–80.
- CAMPOS, C. P. de. *Problemas dinâmicos em geometria computacional*. Dissertação (Mestrado) — Instituto Matemática e Estatística da USP, São Paulo, 2001.
- CAMPOS, C. P. de; COZMAN, F. G. Inference in credal networks using multilinear programming. In: *Proceedings of the Second Starting AI Researcher Symposium*. Valencia: IOS Press, 2004. p. 50–61.
- CAMPOS, C. P. de; COZMAN, F. G. Belief updating and learning in semi-qualitative probabilistic networks. In: *Conference on Uncertainty in Artificial Intelligence*. Edinburgh, UK, 2005. p. 153–160.
- CAMPOS, C. P. de; COZMAN, F. G. Computing lower and upper expectations under epistemic independence. In: *Fourth International Symposium on Imprecise Probabilities and Their Applications*. Pittsburgh, USA, 2005. p. 78–87.

- CAMPOS, C. P. de; COZMAN, F. G. The inferential complexity of Bayesian and credal networks. In: *International Joint Conference on Artificial Intelligence*. Edinburgh, UK: IJCAI, 2005. p. 1313–1318.
- CAMPOS, L. de; MORAL, S. Independence concepts for convex sets of probabilities. In: *XI Conference on Uncertainty in Artificial Intelligence*. San Francisco: Morgan Kaufmann, 1995. p. 108–115.
- CANO, A.; CANO, J. E.; MORAL, S. Convex sets of probabilities propagation by simulated annealing. In: GOOS, G.; HARTMANIS, J.; LEEUWEN, J. van (Ed.). *Proceedings of the International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Paris, France: IPMU, 1994. p. 978–983.
- CANO, A.; MORAL, S. A genetic algorithm to approximate convex sets of probabilities. In: *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems*. Granada: IPMU, 1996. v. 2, p. 859–864.
- CANO, A.; MORAL, S. Using probability trees to compute marginals with imprecise probabilities. *International Journal of Approximate Reasoning*, v. 29, p. 1–46, 2002.
- CANO, J.; DELGADO, M.; MORAL, S. An axiomatic framework for propagating uncertainty in directed acyclic networks. *International Journal of Approximate Reasoning*, v. 8, p. 253–280, 1993.
- CHARNIAK, E. Bayesian networks without tears. *AI Magazine*, v. 12(4), p. 50–63, 1991.
- CHRISMAN, L. Independence with lower and upper probabilities. In: *XII Conference on Uncertainty in Artificial Intelligence*. San Francisco: Morgan Kaufmann, 1996. p. 169–177.
- COLETTI, G.; SCOZZAFAVA, R. Stochastic independence in a coherent setting. *Annals of Mathematics and Artificial Intelligence*, v. 35, p. 151–176, 2002.
- COUSO, I.; MORAL, S.; P., W. A survey of concepts of independence for imprecise probabilities. *Risk, Decision and Policy*, v. 5, p. 165–181, 2000.
- COZMAN, F. G. Robustness analysis of Bayesian networks with local convex sets of distributions. In: *XIII Conference on Uncertainty in Artificial Intelligence*. San Francisco, California: Morgan Kaufmann, 1997. p. 108–115.
- COZMAN, F. G. Credal networks. *Artificial Intelligence*, v. 120, p. 199–233, 2000.
- COZMAN, F. G. Generalizing variable elimination in Bayesian networks. In: *Workshop on Probabilistic Reasoning in Artificial Intelligence*. São Paulo, Brasil: Escola Politécnica, USP, 2000. p. 27–32.
- COZMAN, F. G. Separation properties of sets of probabilities. In: BOUTILIER, C.; GOLDSZMIDT, M. (Ed.). *XVI Conference on Uncertainty in Artificial Intelligence*. San Francisco: Morgan Kaufmann, 2000. p. 107–115.

- COZMAN, F. G. Algorithms for conditioning on events of zero lower probability. In: *Proc. of the Fifteenth Int. Florida Artificial Intelligence Research Society Conf.* Pensacola, Florida: FLAIRS, 2002. p. 248–252.
- COZMAN, F. G.; CAMPOS, C. P. de. Local computation in credal networks. In: *Proceedings of the Workshop on Local Computation for Logics and Uncertainty.* Valencia: IOS Press, 2004. p. 5–11.
- COZMAN, F. G.; ROCHA, J. C. F. da; CAMPOS, C. P. de. Computing with sets of probability measures. In: *SIAM Workshop on Validated Computing.* Toronto, Canadá. 2002. p. 45–48.
- DECHTER, R. Bucket elimination: A unifying framework for probabilistic inference. In: HORVITZ, E.; JENSEN, F. (Ed.). *Proceedings of the XII Conference on Uncertainty in Artificial Intelligence (UAI-96).* San Francisco: Morgan Kaufmann Publishers, 1996. p. 211–219. ISBN 1-55860-412-X.
- DEMPSTER, A. P. A generalization of Bayesian inference. *Journal of the Royal Statistical Society Series B*, v. 30, p. 205–247, 1968.
- DRUZDZEL, M. J.; GAAG, L. C. van der. Elicitation of probabilities for belief networks: Combining qualitative and quantitative information. In: *UAI*, 1995. p. 141–148.
- DRUZDZEL, M. J.; HENRION, M. Belief propagation in qualitative probabilistic networks. In: CARRETE, N. P.; SINGH, M. (Ed.). *Qualitative Reasoning and Decision Technologies.* CIMNE, Barcelona, 1993. p. 451–460.
- DUFFIN, R. J. Linearizing geometric problems. *SIAM Review*, v. 12(211), 1970.
- EYAL, A. *Dividing and Conquering Logic.* Tese (Doutorado) — Stanford University, 2001.
- FAGIUOLI, E.; ZAFFALON, M. 2U: An exact interval propagation algorithm for poly-trees with binary variables. *Artificial Intelligence*, v. 106(1), p. 77–107, 1998.
- FORTNOW, L.; REINGOLD, N. PP is closed under truth-table reductions. In: *Structure in Complexity Theory Conference*, 1991. p. 13–15.
- GAAG, L. C. van der. Computing probability intervals under independency constraints. In: BONISSONE, P. P.; HENRION, M.; LEMMER, J. F. (Ed.). *VI Conference on Uncertainty in Artificial Intelligence.* Amsterdam: Elsevier Science, 1991. p. 457–466.
- GAAG, L. C. van der; BODLAENDER, H. L.; FEELDERS, A. Monotonicity in Bayesian networks. In: *Proceedings of the 20th conference on Uncertainty in artificial intelligence.* Banff, Canadá: AUAI Press, 2004. p. 569–576. ISBN 0-9749039-0-6.
- GEBHARDT, J.; KRUSE, R. Learning possibilistic networks from data. In: FISHER, D.; LENZ, H. J. (Ed.). *International Workshop on Artificial Intelligence and Statistics.* Fort Lauderdale/New York: Springer-Verlag, 1995. p. 233–243.
- GOCHET, W.; SMEERS, Y. A branch-and-bound method for reversed geometric programming. *Operations Research*, v. 27(5), p. 983–996, 1979.

- HA, V. A. Geometric foundations for interval-based probabilities. *Annals of Mathematics and Artificial Intelligence*, v. 24(1-4), p. 1–21, 1998.
- HAARHOFF, P. C.; BUYS, J. D. A new method for the optimization of a nonlinear function subject to nonlinear constraints. *Comp. J.*, v. 13(178), 1970.
- HAILPERIN, T. *Sentential Probability Logic*. Bethlehem: Lehigh University Press, 1996.
- HANSEN, P.; JAUMARD, B. Probabilistic satisfiability. In: *Technical Report G-96-31, Les Cahiers du GERAD*. École Polytechnique de Montréal, 1996.
- HORST, R.; TUY, H. *Global Optimization: Deterministic Approaches*. Heidelberg: Springer Verlag, 1996.
- IDE, J. S.; COZMAN, F. G. Random generation of Bayesian networks. In: *Brazilian Symposium on Artificial Intelligence*. Recife: Springer-Verlag, 2002. p. 366–375.
- JENSEN, F. V. *An Introduction to Bayesian Networks*. New York: Springer Verlag, 1996.
- KAKADE, S.; ORTIZ, L. Graphical economics. *COLT*, 2004.
- KIKUTI, D.; COZMAN, F. G.; CAMPOS, C. P. de. Partially ordered preferences in decision trees: Computing strategies with imprecision in probabilities. In: *IJCAI Workshop about Advances on Preference Handling*. Edinburgh, UK, 2005.
- KLOKS, T. *Treewidth: computations and approximations*. Springer-Verlag, 1994. (Lecture Notes in Computer Science, v. 842).
- KYBURG JR., H. E. Higher order probabilities. In: *UAI*. Elsevier, 1987. p. 15–22.
- LEVI, I. *The Enterprise of Knowledge*. Cambridge, Massachusetts: MIT Press, 1980.
- LITTMAN, M. L.; GOLDSMITH, J.; MUNDHENK, M. The computational complexity of probabilistic planning. *Journal of Artificial Intelligence Research*, v. 9, p. 1–36, 1998.
- LUKATSKII, A. M.; SHAPOT, D. V. Problems in multilinear programming. *Computational Mathematics and Mathematical Physics*, v. 41(5), p. 638–648, 2000.
- LUO, C.; YU, C.; LOBO, J.; WANG, G.; PHAM, T. Computation of best bounds of probabilities from uncertain data. *Computational Intelligence*, v. 12(4), p. 541–566, 1996.
- MARANAS, C. D.; FLOUDAS, C. A. Global optimization in generalized geometric programming. *Computers and Chemical Engineering*, v. 21(4), p. 351–370, 1997.
- NILSSON, N. J. Probabilistic logic. *Artificial Intelligence*, v. 28, p. 71–87, 1986.
- PAPADIMITRIOU, C. H. *Computational Complexity*. San Diego: Addison-Wesley, 1994.
- PARDALOS, P. M.; ROSEN, J. B. *Constrained Global Optimization: Algorithms and Applications*. Berlin/Heidelberg/New York: Springer Verlag, 1987. (Lecture Notes in Computer Science, v. 268).

- PARK, J. D. MAP complexity results and approximation methods. In: DARWICHE, A.; FRIEDMAN, N. (Ed.). *Proceedings of the XVIII Conference in Uncertainty in Artificial Intelligence*. University of Alberta, Edmonton, Alberta, Canadá: Morgan Kaufmann, 2002. ISBN 1-55860-897-4.
- PARK, J. D.; DARWICHE, A. Complexity results and approximation strategies for MAP explanations. *Journal of Artificial Intelligence Research*, v. 21, p. 101–133, 2004.
- PARSONS, S.; DOHNAL, M. A semiquantitative approach to reasoning in probabilistic networks. *Applied Artificial Intelligence*, v. 7, p. 223–235, 1993.
- RATNER, M.; LASDON, L. S.; JAIN, A. Solving geometric problems using GRG: Results and comparisons. *JOTA*, v. 26(253), 1978.
- RENOOIJ, S.; GAAG, L. C. van der. Enhancing QPNs for trade-off resolution. In: *UAI*, 1999. p. 559–566.
- RENOOIJ, S.; GAAG, L. C. van der. Exploiting non-monotonic influences in qualitative belief networks. In: *Proceedings of the Eighth International Conference on Information Processing and Management of Uncertainty in Knowledge-based Systems*. Madrid, Spain, 2000. p. 1285–1290.
- RENOOIJ, S.; GAAG, L. C. van der. From qualitative to quantitative probabilistic networks. In: *Proceedings of the 18th Annual Conference on Uncertainty in Artificial Intelligence (UAI-02)*. San Francisco, CA: Morgan Kaufmann Publishers, 2002. p. 422–429.
- RENOOIJ, S.; GAAG, L. C. van der; PARSONS, S. Propagation of multiple observations in qualitative probabilistic networks. In: *Proceedings of the 12th Belgium-Netherlands Conference on Artificial Intelligence*, 2000. p. 235–242.
- RENOOIJ, S.; GAAG, L. C. van der; PARSONS, S. Context-specific sign-propagation in qualitative probabilistic networks. *Artif. Intell.*, Elsevier Science Publishers Ltd., v. 140, n. 1-2, p. 207–230, 2002. ISSN 0004-3702.
- RIJCKAERT, M. J.; MARTENS, X. M. Comparison of generalized geometric programming algorithms. *JOTA*, v. 26(205), 1978.
- ROCHA, J. C. da; COZMAN, F. G.; CAMPOS, C. P. de. IDS: A divide-and-conquer algorithm for inference in polytree-shaped credal networks. In: *Encontro Nacional de Inteligência Artificial*. Sao Leopoldo, Brasil, 2005. p. 553–562.
- ROCHA, J. C. F. da; COZMAN, F. G. Inference with separately specified sets of probabilities in credal networks. In: *XVIII Conference on Uncertainty in Artificial Intelligence*. San Francisco: Morgan Kaufmann, 2002. p. 430–437.
- ROCHA, J. C. F. da; COZMAN, F. G.; CAMPOS, C. P. de. Inference in polytrees with sets of probabilities. In: *XIX Conference on Uncertainty in Artificial Intelligence*. New York: Morgan Kaufmann, 2003. p. 217–224.
- ROTH, D. On the hardness of approximate reasoning. *Artificial Intelligence*, v. 82, n. 1-2, p. 273–302, 1996.

- RUSSELL, S.; BINDER, J.; KOLLER, D.; KANAZAWA, K. Local learning in probabilistic networks with hidden variables. In: *XIV International Joint Conference on Artificial Intelligence*. Montreal, Canada: Morgan Kaufmann, 1995. p. 1146–1152.
- RUSSELL, S.; NORVIG, P. *Artificial Intelligence: A Modern Approach*. Upper Saddle River: Prentice Hall, 2002.
- SHAFER, G. *A Mathematical Theory of Evidence*. Princeton: Princeton University Press, 1976.
- SHAFER, G. Perspectives on the theory and practice of belief functions. *International Journal of Approximate Reasoning*, v. 3, p. 1–40, 1990.
- SHAFER, G. *Probabilistic Expert Systems*. Philadelphia: SIAM, 1996. (CBMS-NSF regional conference series in applied mathematics, v. 67).
- SHERALI, H. D.; ADAMS, W. P. *A Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems*. Dordrecht/Boston/London: Kluwer Academic Publishers, 1999.
- SHERALI, H. D.; TUNCBILEK, C. H. A global optimization algorithm for polynomial programming problems using a reformulation-linearization technique. *Journal of Global Optimization*, v. 2, p. 101–112, 1992.
- SHIMONY, S. E. Finding MAPs for belief networks is NP-hard. *Artificial Intelligence*, v. 68(2), p. 399–410, 1994.
- TESSEM, B. Interval probability propagation. *International Journal of Approximate Reasoning*, v. 7, p. 95–120, 1992.
- TUY, H. *Convex Analysis and Global Optimization*. Dordrecht: Kluwer Academic Publishers, 1998. (Nonconvex Optimization and Its Applications, v. 22).
- VANTAGGI, B. Graphical representation of asymmetric graphoid structures. In: *Third Int. Symp. on Imprecise Probabilities and Their Applications*. Lugano: Carleton Scientific, 2003. p. 560–574.
- WALLEY, P. *Statistical Reasoning with Imprecise Probabilities*. London: Chapman and Hall, 1991.
- WALLEY, P. Measures of uncertainty in expert systems. *Artificial Intelligence*, v. 83, p. 1–58, 1996.
- WELLMAN, M. P. Fundamental concepts of qualitative probabilistic networks. *Artificial Intelligence*, Elsevier Science Publishers Ltd., v. 44, n. 3, p. 257–303, 1990. ISSN 0004-3702.
- ZAFFALON, M. *Inferenze e Decisioni in Condizioni di Incertezza con Modelli Grafici Orientati*. Tese (Doutorado) — Università di Milano, 1997.
- ZAFFALON, M.; FAGIUOLI, E. Tree-based credal networks for classification. *Reliable Computing*, v. 9(6), p. 487–509, 2003.