

PUC

Série: Monografias em Ciência da Computação,
No. 29/89

ALGORITMOS PARA COMBINAÇÃO DE EVIDÊNCIAS EM ESPAÇO
HIERARQUIZADO DE PROPOSIÇÕES

Wagner T. Silva
Ruy L. Milidiú

Departamento de Informática

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO DE JANEIRO
RUA MARQUÊS DE SÃO VICENTE, 225 - CEP-22453
RIO DE JANEIRO - BRASIL

PUC/RJ - DEPARTAMENTO DE INFORMATICA

Séries: Monografias em Ciências da Computação, Nº 29/89.

Editor: Paulo A. S. Veloso

Dezembro, 1989.

ALGORITMOS PARA COMBINAÇÃO DE EVIDÊNCIAS
EM ESPAÇO HIERARQUIZADO DE PROPOSIÇÕES*

por

Wagner Teixeira da Silva
DI-PUC/RJ e CIC-UNB

Ruy Luiz Milidiú
DI-PUC/RJ e IM-UFRJ

* Este trabalho foi parcialmente financiado pela CAPES e FINEP.

Responsável por publicações:

Rosane Teles Lins Castilho
Assessoria de Biblioteca, Documentação e Informação
PUC RIO, Departamento de Informática
Rua Marquês de São Vicente, 225 - Gávea
22453 - Rio de Janeiro, RJ
BRASIL

Tel.: (021) 529-9386
BITNET: usertrlc@lncc.bitnet

TELEX: 31078

FAX: (021) 274-4546

RESUMO

Este artigo propõe dois algoritmos, um paralelo e um sequencial, para combinação de evidências em um espaço hierarquizado de proposições. A representação de incertezas segue o enfoque da teoria de Dempster e Shafer. São resumidos os tópicos relevantes dessa teoria para a apresentação dos algoritmos. A definição de hierarquia de hipóteses é apresentada, sendo também exploradas suas propriedades quanto à estruturação de incertezas. Para o projeto dos algoritmos é aplicado o paradigma de orientação a objetos.

PALAVRAS-CHAVE: teoria de Dempster e Shafer, teoria da evidência, funções de crença, hierarquia de hipóteses, hierarquia de proposições, combinação de evidências, quadro de discernimento, paradigma de orientação a objetos.

ABSTRACT

This paper describes a parallel and a sequential algorithm for evidence combination on a hierarchical proposition space. We assume Dempster-Shafer Theory approach to uncertainty modeling. Hence, we also summarize all relevant aspects of this theory necessary on both algorithms description. Some important properties of hierarchical proposition spaces are shown in order to derive our computational schemes. The design of the algorithms is object oriented.

KEY-WORDS: Dempster-Shafer theory, evidence theory, belief functions, hierarchical hypothesis, hierarchical propositions, frame of discernment, object-oriented paradigm.

SUMÁRIO

	página
1. INTRODUÇÃO	01
2. TEORIA DE DEMPSTER E SHAFER	
2.1 CRENÇA	02
2.2 MASSA BÁSICA DE CRENÇA	06
2.3 TIPOS DE FUNÇÕES CRENÇA	07
2.4 QUADROS DE DISCERNIMENTO	08
3. ORIENTAÇÃO A OBJETOS	
3.1 CONCEITO DE OBJETO	15
3.2 CONCEITO DE CLASSE	15
3.3 SISTEMA DE SUPORTE	16
4. COMBINAÇÃO DE EVIDÊNCIAS HIERARQUIZADAS	
4.1 HIERARQUIA DE PROPOSIÇÕES	17
4.2 EVIDÊNCIAS	19
4.3 FLUXO DE INFORMAÇÃO ENTRE OS NÓS	20
4.4 PROPRIEDADES ESTRUTURAIS	21
4.5 CÁLCULO DE CRENÇAS	23
5. ALGORITMO SEQUENCIAL	
5.1 MOTIVAÇÃO	33
5.2 SUPOSIÇÕES	33
5.3 NOTAÇÃO	33
5.4 OBJETIVO E RESUMO	34
5.5 ENTRADA E SAÍDA	34
5.6 PSEUDO-CÓDIGO	35
5.7 ANÁLISE DE COMPLEXIDADE	38
6 ALGORITMO PARALELO	
6.1 EXPLICITANDO O PARALELISMO	41
6.2 PARADIGMA DE OBJETOS	42
6.3 DADOS E OPERAÇÕES POR CLASSE DE NÓS	42
6.4 ANÁLISE DE COMPLEXIDADE	50
7 CONCLUSÕES	53
REFERÊNCIAS	54

INTRODUÇÃO

No contexto da Teoria da Evidência, este artigo propõe dois algoritmos, um paralelo e um seqüencial, para solução do problema da hierarquia de proposições, proposto por Gordon e Shortliffe(1985). Os dois algoritmos usam os resultados do trabalho de Barnett(1981), o que permite reduzir a complexidade do tempo de computação para linear.

Um algoritmo seqüencial já foi proposto por Shafer e Logan(1987) para solução deste problema. Eles afirmam que esse algoritmo é linear, a despeito da "subrotine 5" do algoritmo ter complexidade quadrática no fator de ramificação da árvore de hierarquia. Esta controvérsia justifica a proposta de um algoritmo seqüencial neste artigo. A versão paralela aqui proposta se justifica por visar um contexto distribuído e dinâmico para coleta e combinação de evidências, bem como para tomada de decisões. A linearidade do esforço computacional na quantidade de evidências disponíveis e no tamanho do quadro de discernimento do problema é preservada também na versão distribuída.

A seção 2 resume os principais tópicos da teoria de DS usados neste artigo. A seção 3 resume o paradigma de orientação a objetos. A seção 4 apresenta o problema de hierarquia de proposições e suas propriedades. Na seção 5, é formulado o algoritmo seqüencial para combinação de evidências em um espaço de proposições hierarquizado e analisada sua complexidade em tempo e espaço. Na seção 6, é formulado o algoritmo paralelo e analisada sua complexidade. A seção 7, conclui o presente texto com comentários gerais sobre o processo de combinar evidências.

2 TEORIA DE DEMPSTER E SHAFER

A Teoria da Evidência de Shafer(1976) resultou da reformulação do trabalho de Dempster(1963, 1967a, 1967b, 1967c, 1968a, 1968b, 1969) sobre limites inferior e superior de probabilidades, daí o fato desta teoria ser conhecida, também, como Teoria de Dempster-Shafer (DS). Esta reformulação foi feita em termos de domínios discretos finitos, comumente encontrados na área de sistemas baseados em conhecimento, e expressa em termos mais relevantes tais como crença, suporte e evidência. Contudo, o coração da Teoria da evidência é a regra de combinação de

Dempster, ou soma ortogonal. Os conceitos de limites inferior e superior de probabilidade são identificados como graus de crença e graus de plausibilidade no trabalho de Shafer. O nome "Teoria da Evidência" decorre do fato dela tratar com pesos de evidência e graus numéricos de suporte baseados em evidências. A seguir são resumidos os tópicos da teoria de DS usados neste artigo. O leitor interessado em mais detalhes deve consultar as seguintes referências: Shafer(1976, 1986), que aborda de forma exaustiva toda a teoria; ou Silva e Milidiu(1989), que usam um enfoque mais acessível para uma leitura introdutória; ou ainda Shafer e Logan(1987), Smets(1988), Shatnagar e Kanal(1986), Gordon e Shortliffe(1984, 1985), Lemmer(1986), Barnett(1981) e Senne(1987), que são boas referências para situar a teoria no contexto das ferramentas usadas para representar e atualizar o conhecimento impreciso.

2.1 CRENÇA

O termo crença significa o ato ou efeito de crer; ter fé, convicção, etc. Portanto, é algo epistêmico, subjetivo e pode variar de pessoa para pessoa e também com o tempo. Há, ainda, gradação de crença. Alguém pode acreditar muito, pouco ou nada em uma determinada proposição.

Na Teoria da evidência, um agente em contato com o mundo, coleta informações, e para cada peça de evidência ele faz um julgamento, subjetivo ou não, do quanto aquela evidência suporta sua crença em proposições de um dado conjunto finito de proposições Θ , das quais é desconhecida a única verdadeira. Este conjunto Θ é chamado de quadro de discernimento, e cada subconjunto A de Θ representa uma disjunção das suas proposições atômicas. Logo cada $A \subset \Theta$ é uma proposição também. O grau com que uma evidência suporta a crença de alguém em uma disjunção A de Θ é um valor do intervalo $[0,1]$. Ter crença igual a 1 em determinada proposição significa completa certeza. O Exemplo 1.1 ilustra estes conceitos.

Exemplo 1.1: Suponha, por exemplo, que em uma eleição presidencial existam quatro candidatos: A , B , C e D . Obviamente apenas um será o vencedor. Seja θ_x a proposição afirmando a vitória do candidato

x , e $\Theta = \{\theta_A, \theta_B, \theta_C, \theta_D\}$. Um agente, inicialmente, nada sabendo sobre os candidatos, nem tampouco sobre os seus respectivos partidos poderá representar sua ignorância atribuindo uma crença igual a 0 a cada proposição do conjunto Θ , e igual a 1 ao conjunto Θ . Esta atribuição significa que uma das proposições é necessariamente verdadeira, não se tendo, ainda, nenhuma evidência que favoreça uma particular proposição de Θ . Posteriormente, após coletar informações sobre os candidatos, então a crença do agente poderá ser atualizada, refletindo as evidências pró e contra os diversos candidatos.

Cada peça de evidência determina uma função de crença. Assim, dado um corpo de evidências ξ , fica determinado, por julgamento, um conjunto de funções de crença ζ . A cada peça de evidência de ξ está associada uma função de crença de ζ . As funções de crenças de ζ são combinadas gerando uma função de crença atualizada que é suportada pelo corpo de evidências ξ . Havendo mais de um agente envolvido no processo, a função de crença final, representa o consenso lógico dado pela regra de combinação de evidências, a qual é conhecida como soma ortogonal ou Regra de Dempster.

A massa de crença unitária distribuída entre as disjunções A de Θ , com base num corpo de evidências ξ , que é específica de cada A , e não partilhada por nenhum dos seus subconjuntos próprios, é dado por uma distribuição de probabilidade P da variável aleatória x , tomando valores no conjunto das partes de Θ . Isto é:

- $P[x=A]=y, y \in [0,1]$ e $A \subset \Theta$
- $P[x=\phi]=0,$
- $\sum_{A \subset \Theta} P[x=A] = 1$

Dentro do enfoque evidencial, contudo, $m(A)=P[x=A]$, é conhecido como probabilidade básica associada ou massa básica de crença (mbc). A mbc $m: 2^\Theta \rightarrow [0,1]$ deve obedecer as regras:

$$\bullet m(\phi) = 0 \quad (1.1)$$

$$\bullet \sum_{A \subset \Theta} m(A) = 1 \quad (1.2)$$

Dá-se o nome de elementos focais aos subconjuntos $A \subset \Theta$, tais que $m(A) > 0$, e o conjunto F de tais elementos é chamado conjunto

focal. A cada mbc m está associada uma função de crença Cr e vice-versa. Uma função $Cr: 2^{\Theta} \rightarrow [0,1]$, que atribui graus de crença a cada subconjunto A de Θ , é chamada de função de crença se, e só se, existe um subconjunto aleatório x de Θ , não vazio, tal que

$$\begin{aligned} Cr(A) &= Pr[x \in A], \\ &= \sum_{B \subset A} Pr[x=B] \\ &= \sum_{B \subset A} m(B), \quad \forall A \subset \Theta \end{aligned} \quad (1.3)$$

Duas funções de crença Cr_1 e Cr_2 podem ser combinadas em termos dos seus conjuntos aleatórios ou equivalentemente de suas mbc. Sejam x_1, x_2 subconjuntos aleatórios não vazios do quadro de discernimento Θ . Suponha que x_1 e x_2 sejam probabilisticamente independentes, e que $Pr[x_1 \cap x_2 \neq \emptyset] > 0$. Sejam m_1 e m_2 as mbc de Cr_1 e Cr_2 , respectivamente, dadas por

$$\begin{aligned} m_1(A_1) &= Pr[x_1 = A_1], \quad \forall A_1 \subset \Theta; \text{ e} \\ m_2(A_2) &= Pr[x_2 = A_2], \quad \forall A_2 \subset \Theta. \end{aligned}$$

Então a crença $Cr = Cr_1 \oplus Cr_2$, resultante da combinação de Cr_1 e Cr_2 , tem sua mbc m dada por

$$m(A) = Pr[x_1 \cap x_2 = A \mid x_1 \cap x_2 \neq \emptyset], \quad \forall A \subset \Theta$$

Na proposta de Shafer (1976) a função de crença $Cr: 2^{\Theta} \rightarrow [0,1]$ goza das seguintes propriedades:

- a.1 $Cr(\emptyset) = 0$
- a.2 $Cr(\Theta) = 1$
- a.3 Se A_1, \dots, A_n são subconjuntos de Θ , então

$$Cr\left(\bigcup_{i=1}^n A_i\right) \geq \sum_{I \subset \{1, \dots, n\}} (-1)^{|I|+1} Cr\left(\bigcap_{i \in I} A_i\right)$$

- a.4 Sejam Cr_1 e Cr_2 funções de crenças, Cr_1 com mbc m_1 e elementos focais A_1, \dots, A_m ; e Cr_2 com mbc m_2 e elementos focais B_1, \dots, B_n .

$$\text{Se } K^{-1} = \sum_{A_i \cap B_j \neq \emptyset} m_1(A_i) \cdot m_2(B_j) > 0, \text{ então}$$

$$m(x) = K \sum_{\substack{A_i, B_j \\ x}} m_1(A_i) \cdot m_2(B_j) ,$$

Onde m é uma mbc e a sua função de crença Cr é o resultado da combinação de Cr_1 e Cr_2 , expressa como $Cr = Cr_1 \oplus Cr_2$, e chamada de Soma Ortogonal ou Regra de Dempster.

A regra a.1 diz que nenhuma crença é atribuída ao conjunto vazio. Ela é demonstrável a partir de (1.1) e (1.3).

A regra a.2 diz que uma das proposições de Θ é certamente verdadeira. Isto é lógico, desde que Θ contém todas as possíveis respostas ao problema, das quais apenas uma é verdadeira: este é o significado de exaustivo e exclusivo atribuído ao conjunto de proposições Θ . Esta regra segue-se das fórmulas (1.2) e (1.3).

A regra a.3 é o grande trunfo da função de crença. Ela permite representar a ignorância. Isto é, quando parte da crença do agente, não tem suporte das evidências para nenhuma das proposições de um dado subconjunto, então esta parte de crença é atribuída ao subconjunto, e é específica dele. Isto foi ilustrado no Exemplo 1.1, onde, face a ausência de evidências, toda a massa de crença foi atribuída ao próprio Θ . Agora, suponha que o agente toma conhecimento de que os candidatos A e B são de direita, e os candidatos C e D são de esquerda. Em uma pesquisa de opinião pública há evidência que favorece os candidatos de direita em 30%, sendo que os demais 70% estão indefinidos. Esta evidência suporta a função de crença Cr_1 , cuja mbc é dada abaixo:

$$m_1((A,B))=0.3, \text{ e } m_1(\Theta)=0.7 \quad (1.4)$$

Posteriormente, o agente toma conhecimento de uma campanha difamatória contra o candidato B , levando a um contexto favorável as candidaturas de esquerda. Agora, as evidências suportam uma outra função de crença Cr_2 , com mbc dada abaixo:

$$m_2((C,D))=0.4, \text{ e } m_2(\Theta)=0.6 \quad (1.5)$$

Estas duas peças de evidências suportando Cr_1 e Cr_2 ilustram a representação da ignorância facultada pela regra a.3. Em síntese, esta regra diz que a crença no todo é maior ou igual a soma das crenças nas partes.

A regra a.4 permite atualizar a crença nas proposições de Θ sempre que novas evidências se tornem disponíveis. Ela também estabelece quando duas funções de crença são combináveis. A combinação de duas funções de crença está definida somente quando existirem pelo menos dois elementos focais, um de cada uma das funções, cuja interseção for não vazia. Isto é, o conflito não é absoluto. Combinando as crenças Cr_1 e Cr_2 obtém-se $Cr = Cr_1 \oplus Cr_2$ com mbc m dada por

$$m(\{A, B\}) = 0.20, \quad m(\{C, D\}) = 0.32 \quad \text{e} \quad m(\Theta) = 0.48$$

A crença resultante Cr atribui 20% de chance para a Direita, 32% para a Esquerda e 48% está indefinido. Estes 48% representam a ignorância do agente. A vista de mais evidências esta ignorância pode diminuir pela transferência de parte da crença alocada em Θ para os seus subconjuntos próprios.

A regra a.4 também está definida para um conjunto finito de funções de crença, C_1, C_2, \dots, C_n , se estiver definida para qualquer par distinto de funções de crença C_i, C_j , com $i, j \in \{1, 2, \dots, n\}$, e $i \neq j$. A soma $C_1 \oplus C_2 \oplus \dots \oplus C_n$ pode ser realizada pela aplicação sucessiva da regra de Dempster. E é fácil mostrar que a soma ortogonal assim definida é associativa.

O nome "Soma Ortogonal" lembra que as evidências suportando as funções a serem combinadas devem vir de fontes distintas. Sem o que o processo de combinação pode gerar enviesamento.

2.2 MASSA BÁSICA DE CRENÇA

Há diversas formas de representar uma função de crença Cr , (Silva e Milidiú, 1989) todas convertíveis entre si. A representação mais compacta é através da função de massa básica de crença, mbc. Esta representação é dada por $\mathcal{M} = (\Theta, \mathbb{F}, \mu)$. Onde

- Θ - é o quadro de discernimento, Conjunto exaustivo de proposições mutuamente exclusivas.
- \mathbb{F} - é o conjunto dos elementos focais de Cr , isto é:

$$\mathbb{F} = \{A \subseteq \Theta \mid m(A) > 0\},$$
o qual é chamado de conjunto focal.
- μ - é uma restrição da função m ao conjunto focal \mathbb{F} , isto é, $\mu: \mathbb{F} \rightarrow (0, 1]$ e $\mu(A) = m(A)$, $\forall A \in \mathbb{F}$.

Chamamos de núcleo da função de crença Cr ao conjunto N definido por

$$N = \bigcup_{x \in F} x.$$

No núcleo está concentrada toda a massa de crença da função Cr, isto é $Cr(N) = 1$.

Outras medidas de crença, além da mbc m e função de crença Cr, importantes para caracterizar melhor a crença de alguém nas proposições do quadro de discernimento Θ são

- Pl - Função de Plausibilidade. Atribui a cada subconjunto A de Θ a massa de crença $Pl(A)$, Plausibilidade de A, que não contradiz a crença em A, isto é, $Pl(A) = 1 - Cr(\bar{A})$.
- Q - Função comunitária. Atribui a cada subconjunto A de Θ a massa de crença $Q(A)$ que está livre para migrar para os elementos de A.

Dada uma função de crença na representação \mathcal{M} , pode-se obter as outras medidas de crença em termos da função μ . As fórmulas de conversão de \mathcal{M} para a função de crença Cr, plausibilidade Pl, e função comunitária Q são as seguintes:

$$Cr(A) = \sum_{\substack{B \subset A, \\ B \in F}} \mu(B), \quad \forall A \subset \Theta \quad (2.1)$$

$$Pl(A) = \sum_{\substack{A \cap B \neq \emptyset, \\ B \in F}} \mu(B), \quad \forall A \subset \Theta \quad (2.2)$$

$$Q(A) = \sum_{\substack{B \subset F, \\ A \subset B}} \mu(B), \quad \forall A \subset \Theta \quad (2.3)$$

2.3 TIPOS DE FUNÇÕES DE CRENÇA

As funções de crença se dividem em dois grandes grupos: funções de suporte e funções não-suporte. Cada um destes grupos também se subdividem. A estrutura apresentada por cada grupo ou subgrupo pode ser explorada no processo de combinação de evidências.

(a) Função de suporte

Seja Cr uma função de crença, e sejam m e N, respectivamente, a mbc e o núcleo associados a Cr. Se $m(N) > 0$, então dizemos que Cr é uma particular função de crença chamada de função de

suporte.

Exemplo: Sejam $\Theta = (a,b,c)$, Cr com mbc dada por $m(\{a\})=1/5$ e $m(\{a,b\})=4/5$. O conjunto $\{a,b\}$ é o núcleo de Cr. O fato deste ter mbc maior que zero torna Cr uma função de suporte.

(b) Função de suporte simples (fss).

Uma função de suporte Cr é uma fss se $F \subset (A, \Theta)$. Isto é, Cr tem no máximo um elemento focal A distinto de Θ , onde $m(A)=s$ e $m(\Theta)=1-s$, com $0 < s \leq 1$.

Exemplo: A função Cr definida em $\Theta = (a,b,c)$, com $m(\Theta)=0.1$ e $m(\{b,c\})=0.9$.

(c) Função de suporte vacuosa.

Uma função de suporte é vacuosa se $F = \{\Theta\}$. Isto é, o seu único elemento focal é Θ .

Exemplo: A função Cr com $m(\Theta)=1$ e $m(A)=0$, $\forall A \subset \Theta$ e $A \neq \Theta$.

(d) Função dicotômica

Uma função de crença Cr é dicotômica se $F \subset (A, \bar{A}, \Theta)$, $A \subset \Theta$ e $A \neq \Theta$. Isto é, a função Cr não tem outros elementos focais além dos elementos do conjunto (A, \bar{A}, Θ) . As funções vacuosas e de suporte simples são casos particulares da função dicotômica.

Exemplo: A função Cr definida em $\Theta = (a,b,c)$, com $m(\Theta)=0.3$, $m(\{a\})=0.2$ e $m(\{b,c\})=0.5$.

2.4 QUADROS DE DISCERNIMENTO

Há situações em que o quadro de discernimento Θ , inicialmente estruturado para o problema, não é discriminativo para algumas evidências que se tornam disponíveis, isto é, as evidências não são compatíveis com nenhuma das proposições de Θ . Entretanto estas evidências seriam compatíveis com sub-proposições de algumas proposições de Θ . Nestes casos, uma solução é particionar as proposições de Θ , criando um quadro de discernimento Ω que é um refinamento de Θ . Agora, aquelas evidências podem ser usadas para suportar proposições em Ω , e através de implicações este suporte é induzido sobre proposições em Θ . A seguir, são detalhados o conceito de refinamento, e outros conceitos relacionados.

2.4.1 REFINAMENTO E CONDENSAÇÃO

A função $\omega: 2^{\Theta} \rightarrow 2^{\Omega}$ é uma função de refinamento quando Θ e Ω são conjuntos finitos e a família dos conjuntos $\omega(\theta)$, $\forall \theta \in \Theta$, atende aos seguintes axiomas

- (i) $\omega(\theta) \neq \emptyset$;
- (ii) $\omega(\theta) \cap \omega(\theta') = \emptyset$, se $\theta \neq \theta'$;
- (iii) $\Omega = \bigcup_{\theta \in \Theta} \omega(\theta)$.

Neste caso, Ω é dito um refinamento de Θ , e Θ é dito um condensação de Ω .

A figura 2.1 ilustra a função de refinamento ω , dada por $\omega(\theta_1) = \{a, b, c\}$, $\omega(\theta_2) = \{d\}$, e $\omega(\theta_3) = \{e, f, g\}$.

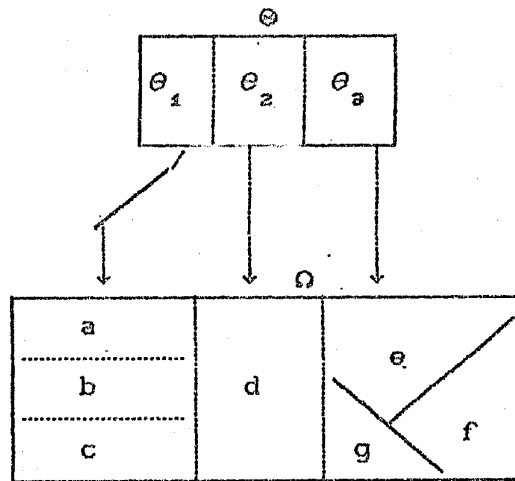


Figura 2.1 Refinamento e condensação

Dada uma partição de Ω , onde cada subconjunto $\omega(\theta)$ é um elemento da partição, podemos afirmar que

$$\omega(A) = \bigcup_{\theta \in A} \omega(\theta), \quad \forall A \subset \Theta$$

Se $\omega: 2^{\Theta} \rightarrow 2^{\Omega}$ é uma função de refinamento, então as seguintes propriedades são verificadas:

- (a) ω é 1-1
- (b) $\omega(\emptyset) = \emptyset$
- (c) $\omega(\Theta) = \Omega$
- (d) $\omega(A \cup B) = \omega(A) \cup \omega(B)$, $\forall A, B \subset \Theta$
- (e) $\omega(\overline{A}) = \overline{\omega(A)}$, $\forall A \subset \Theta$

$$(f) \omega(A \cap B) = \omega(A) \cap \omega(B), \quad \forall A, B \subset \Theta$$

$$(g) \text{ Se } A, B \subset \Theta, \text{ então } \omega(A) \subset \omega(B) \text{ se, e só se, } A \subset B$$

$$(h) \text{ Se } A, B \subset \Theta, \text{ então } \omega(A) \cap \omega(B) = \emptyset, \text{ se, e só se, } A \cap B = \emptyset$$

Combinando o conceito de refinamento ao conceito de redução, descrito no próximo item, obtém-se o ferramental necessário para inferência a partir dos quadros de discernimento refinados para os condensados, e vice-versa.

2.4.2 REDUÇÃO INTERNA E EXTERNA

Dado um refinamento Ω de Θ e um subconjunto w de Ω , e se há interesse em identificar quais subconjuntos de Θ podem implicar ou serem implicados por w , então as funções de redução interna e externa auxiliam nesta tarefa.

A função de redução interna, $\theta^-: 2^\Omega \rightarrow 2^\Theta$, associa à cada subconjunto w de proposições de Ω um subconjunto de proposições A de Θ cujo refinamento está contido em w , isto é,

$$A = \theta^-(w) = \{\theta \in \Theta \mid \omega(\theta) \subset w\}, \quad \forall w \subset \Omega,$$

e neste caso $\omega(A) \subset w$, onde ω é a função de refinamento como definido em 2.4.1.

Exemplo: Na figura 2.1, suponha $w = \{a, b, d\}$. Então $\theta^-(w) = \{\theta_2\}$, e obviamente $\omega(\{\theta_2\}) \subset w$.

A função de redução externa, $\theta^+: 2^\Omega \rightarrow 2^\Theta$, associa à cada subconjunto w de proposições de Ω um subconjunto A de proposições de Θ cujo refinamento de cada elemento de A tem intersecção não vazia com w . Isto é,

$$A = \theta^+(w) = \{\theta \in \Theta \mid \omega(\{\theta\}) \cap w \neq \emptyset\}, \quad \forall w \subset \Omega$$

Exemplo: Na figura 2.1, suponha $w = \{a, b, d\}$. Então $\theta^+(w) = \{\theta_1, \theta_2\}$, $\omega(\{\theta_1\}) \cap w \neq \emptyset$ e $\omega(\{\theta_2\}) \cap w \neq \emptyset$.

As funções acima ensejam o seguinte teorema, cuja demonstração pode ser encontrada em Shafer (1976).

Teorema 2.1:

Sejam $\omega: 2^\Omega \rightarrow 2^\Theta$ a função de refinamento, e θ^- e θ^+ reduções interna e externa para ω . Para todo $w \subset \Omega$ e $A \subset \Theta$, $\omega(A) \subset w$, se, e só se, $A \subset \theta^-(w)$. E também $w \subset \omega(A)$, se, e só se, $\theta^+(w) \subset A$.

Desde que A e $\omega(A)$ correspondem a mesma proposição, $\omega(A) \subset w$ pode ser interpretado como $A \rightarrow w$. Pelo Teorema 2.1, $A \rightarrow w$, se e só se, $A \subset \theta^-(w)$. O subconjunto $\theta^-(w)$ é o maior subconjunto de Θ que implica w . E também, $w \rightarrow A$, se, e só se, $\theta^+(w) \subset A$. O subconjunto $\theta^+(w)$ é o menor subconjunto de Θ que pode ser implicado por w .

Sejam Ω um refinamento de Θ , e Cr uma função de crença sobre Ω , com sua mbc associada m . Então pode-se obter a mbc m_0 sobre Θ , induzida por m , através da fórmula

$$m_0(A) = \sum_{\substack{w \subset \Omega \\ A = \theta^+(w)}} m(w), \quad \forall A \subset \Theta \quad (2.4)$$

2.4.3 DISCERNIMENTO E TRANSFERÊNCIA DE CRENÇA ENTRE QUADROS

As idéias pertinentes a refinamento e condensação levam aos conceitos de funções de crença consistentes, proposições equivalentes, quadros de discernimento compatíveis, quadros de discernimento independentes, e discernimento de evidências. A seguir estes conceitos serão comentados e definidos.

Duas funções de crença definidas sobre dois quadros de discernimento distintos são consistentes se um dos quadros de discernimento é um refinamento do outro, e a crença em uma proposição elementar do quadro condensado é igual a crença da proposição refinada sobre o quadro refinado. A definição a seguir formaliza isto.

Funções consistentes.

Seja Ω um refinamento de Θ , e Cr e Cr_0 funções de crença sobre Ω e Θ , respectivamente, então Cr e Cr_0 são funções de crença consistentes se, e somente se, $Cr_0(A) = Cr(\omega(A))$, para todo $A \in \Theta$.

Exemplo: Considerando o refinamento ilustrado pela figura 2.2, suponha que as funções de crença Cr_1 e Cr_2 estejam definidas sobre Θ e Ω , respectivamente. Onde as mbc de cada uma são dadas por:

$$\begin{aligned} m_1(\theta_1) &= 0.5, \quad m_1(\theta_2) = 0.2, \quad e \quad m_1(\Theta) = 0.3; \quad e \\ m_2(\{a,b\}) &= 0.3, \quad m_2(\{b,c\}) = 0.1, \quad m_2(\{c\}) = 0.1, \quad m_2(\{d\}) = 0.2 \quad e \\ m_2(\Omega) &= 0.3. \end{aligned}$$

É fácil ver que as funções de crença Cr_1 e Cr_2 são consistentes.

Proposições equivalentes.

Seja Ω um refinamento de Θ , com função de refinamento ω . Então para cada proposição $A \in \Theta$ tem-se que A e $\omega(A)$ são proposições equivalentes.

Sejam as proposições $A \in \Theta$ e $w = \omega(A) \in \Omega$, onde Ω é um refinamento de Θ expresso pela função ω . Então A e w são proposições equivalentes e toda a massa de crença existente em A pode ser levada para w ou vice-versa, via as funções de crença consistentes definidas sobre Θ e Ω , respectivamente.

Exemplo: Considere o refinamento ilustrado pela figura 2.1, e suponha que a função de crença Cr , definida sobre Ω , tenha a seguinte mbc:

$$m((a,b))=0.3, m((b,c))=0.1, m((c))=0.1, m((d))=0.2 \text{ e } m(\Omega)=0.3$$

Então a função de crença Cr_1 , sobre Θ , induzida por Cr tem a seguinte mbc:

$$m_1(\theta_1)=0.5, m_1(\theta_2)=0.2, \text{ e } m_1(\Theta)=0.3$$

Obviamente Cr e Cr_1 são funções de crença consistentes.

Dois quadros de discernimento são compatíveis se eles têm um refinamento comum. Dados dois quadros de discernimento, se eles forem compatíveis, então é possível induzir crença de um para o outro. A definição a seguir formaliza este conceito.

Quadros compatíveis.

Dois quadros de discernimento, Θ_1 e Θ_2 são compatíveis se eles têm um refinamento comum Ω . Isto é, existem duas funções ω_1 e ω_2 , tal que $\omega_1: 2^{\Theta_1} \longrightarrow 2^{\Omega}$ e $\omega_2: 2^{\Theta_2} \longrightarrow 2^{\Omega}$.

Se há dois quadros de discernimento compatíveis Θ_1 e Θ_2 , então é possível induzir crenças de proposições de Θ_1 para proposições de Θ_2 , via o refinamento comum Ω , usando as funções de crença consistentes entre Θ_1 e Ω ; e entre Θ_2 e Ω , respectivamente. Pode-se também fazer esta indução de crença entre os quadros consistentes usando as propriedades de implicações apregoadas pelo Teorema 2.1, usando o refinamento comum como escala intermediária entre os dois quadros Θ_1 (fonte) e Θ_2 (destino), ou vice versa, dependendo da direção da implicação desejada.

Exemplo: Sejam $\Theta_1 = (a_1, a_2)$, $\Theta_2 = (b_1, b_2)$, $\Omega = (c_1, c_2, c_3, c_4)$,
 $\omega_1((a_1)) = (c_1, c_2)$, e $\omega_1((a_2)) = (c_3, c_4)$;
 $\omega_2((b_1)) = (c_1, c_2, c_3)$, e $\omega_2((b_2)) = (c_4)$;
 $m_1((a_1)) = 0.4$, $m_1((a_2)) = 0.5$, e $m_1(\Theta_1) = 0.1$;
 $m((c_1, c_2)) = 0.4$, $m((c_3, c_4)) = 0.5$ e $m(\Omega) = 0.1$; e
 $m_2((b_1)) = 0.4$, e $m_2(\Theta_2) = 0.6$

A função de crença Cr_1 , sobre Θ_1 , com a mbc m_1 , induz a função de crença Cr sobre Ω , com mbc m ; e Cr induz a função de crença Cr_2 sobre Θ_2 , com mbc m_2 .

Dois quadros de discernimento compatíveis são independentes se toda função de crença, não vacuosa, sobre um induz uma função de crença vacuosa sobre o outro. A definição abaixo formaliza este conceito.

Quadros independentes.

Dois quadros de discernimento Θ_1 e Θ_2 , compatíveis, são independentes se nenhuma proposição discernida por um não trivialmente implica em uma proposição discernida pelo outro. Em outros termos, suponha que Ω seja um refinamento comum de Θ_1 e Θ_2 , com $\omega_1: 2^{\Theta_1} \rightarrow 2^{\Omega}$ e $\omega_2: 2^{\Theta_2} \rightarrow 2^{\Omega}$. Se $A_1 \subset \Theta_1$, $A_2 \subset \Theta_2$, e $\omega_1(A_1) \subset \omega_2(A_2)$, então $A_1 = \emptyset$ ou $A_2 = \Theta_2$.

Exemplo: Sejam $\Theta_1 = (a_1, a_2)$, $\Theta_2 = (b_1, b_2)$, $\Omega = (c_1, c_2, c_3, c_4)$,
 $\omega_1((a_1)) = (c_1, c_2)$, e $\omega_1((a_2)) = (c_3, c_4)$;
 $\omega_2((b_1)) = (c_1, c_3)$, e $\omega_2((b_2)) = (c_2, c_4)$;
 $m_1((a_1)) = 0.4$, $m_1((a_2)) = 0.5$, e $m_1(\Theta_1) = 0.1$;
 $m((c_1, c_2)) = 0.4$, $m((c_3, c_4)) = 0.5$ e $m(\Omega) = 0.1$; e
 $m_2(\Theta_2) = 1$, e $m_2(A) = 0$, $\forall A \subset \Theta_2, A \neq \Theta_2$.

A função de crença Cr_1 , sobre Θ_1 , com a mbc m_1 , induz a função de crença Cr sobre Ω , com mbc m ; e Cr induz a função de crença vacuosa Cr_2 sobre Θ_2 , com mbc m_2 .

Suponha que Ω seja um refinamento de Θ , e que as evidências sejam coletadas a nível de Ω . Teria o mesmo efeito combinar as evidências sobre Ω e depois induzir o resultado sobre Θ , que,

alternativamente, combinar as crenças parciais induzidas sobre Θ ? O Teorema de interação de evidências, cuja prova pode ser encontrada em Shafer(1976), e os comentários que se seguem procuram esclarecer esta dúvida.

Teorema 2.2, Interação de evidências.

Seja Ω um refinamento de Θ , θ^+ a função de redução externa, e S_1 e S_2 funções de suporte simples sobre Ω , com focos A e B respectivamente. Suponha que $S_1 \oplus S_2$ esteja definida sobre Ω . Então Θ discerne a interação relevante entre S_1 e S_2 , $(S_1|\Theta) \oplus (S_2|\Theta) = (S_1 \oplus S_2)|\Theta$, se, e só se, $\theta^+(A \cap B) = \theta^+(A) \cap \theta^+(B)$. Onde $S|\Theta$ é a função de crença sobre Θ consistente com S sobre Ω .

Quando Θ discerne as evidências coletadas sobre Ω , pode-se fazer toda a combinação sobre Θ e usar Ω apenas como o coletor das evidências: cada evidência coletada e discriminada por Ω , gera uma função de crença C , sobre Ω , a qual induz uma função de crença $C|\Theta$ sobre Θ , as quais são combinadas sobre Θ . Contudo, se Θ não discerne as evidências coletadas e discriminadas por Ω é necessário fazer a coleta e a combinação das evidências sobre Ω , e só então induzir a crença final sobre Θ .

Exemplo: Θ discerne a interação das evidências.

Sejam $\Theta = \{a, b\}$, $\Omega = \{c_1, c_2, c_3, c_4\}$,

$\omega(\{a\}) = \{c_1, c_2\}$, e $\omega(\{b\}) = \{c_3, c_4\}$; e

Cr_1 com mbc dada por $m_1(\{c_1, c_3\}) = 0.4$, e $m_1(\Omega) = 0.6$; e

Cr_2 com mbc dada por $m_2(\{c_3, c_4\}) = 0.5$ e $m_2(\Omega) = 0.5$.

Θ aqui discerne a interação relevante entre Cr_1 e Cr_2 , pois

$\theta^+(\{c_1, c_2\} \cap \{c_3, c_4\}) = \theta^+(\{c_1, c_2\}) \cap \theta^+(\{c_3, c_4\}) = \{b\}$.

Exemplo: Θ não discerne a interação das evidências.

Sejam Θ , Ω e ω como anteriormente, mas Cr_1 e Cr_2 com mbc m_1 e m_2 , respectivamente, definidas como:

$m_1(\{c_1, c_3\}) = 0.4$, e $m_1(\Omega) = 0.6$; e

$m_2(\{c_1, c_4\}) = 0.5$ e $m_2(\Omega) = 0.5$.

O quadro Θ aqui não discerne a interação relevante entre Cr_1 e Cr_2 , pois $Cr_1|\Theta$, $Cr_2|\Theta$ e $(Cr_1|\Theta) \oplus (Cr_2|\Theta)$ são todas funções de crenças vacuossas. Contudo $(Cr_1 \oplus Cr_2)|\Theta$ não é vacuosa. Pode ser

verificado que a restrição imposta para que Θ discirna a interação relevante entre Cr_1 e Cr_2 não foi atendida:

$$\theta^+((c_1, c_2) \cap (c_1, c_4)) = \theta^+(c_1) = (a) \neq \theta^+(c_1, c_2) \cap \theta^+(c_1, c_4) = \emptyset$$

3. ORIENTAÇÃO A OBJETOS

3.1 CONCEITO DE OBJETO

Objeto é tudo aquilo que se apresenta a vista ou ao espírito de alguém. A esta entidade são atribuídas propriedades, ou então verificadas propriedades que lhe são inerentes. Pode-se ver objetos como entidades autônomas. Para cada objeto há um conjunto de possíveis estados internos (não visíveis de fora). Um objeto, num dado instante, está num destes estados internos. A cada objeto está associado um conjunto de operações. Um objeto pode mudar de um estado interno para outro unicamente através das suas próprias operações. Nenhum objeto tem a capacidade de alterar, diretamente, o estado interno de outro objeto. Os objetos se comunicam através de mensagens que é a única forma de interação entre eles. Assim um objeto muda de estado em função do contexto externo dado pela comunidade de objetos a qual ele pertença.

3.2 CONCEITO DE CLASSE

Uma classe é um conjunto de pessoas ou coisas (concretas ou abstratas) com características comuns. E cada objeto será um exemplar de uma dada classe. Aqui classe será definida por um gabarito de dados e serviços. Um gabarito de serviço é a descrição de um procedimento. O conjunto de serviços gabaritado numa particular classe caracteriza o comportamento possível de cada objeto exemplar dessa classe. Quando um objeto x , exemplar da classe X , é criado ele passa a ser uma entidade autônoma com memória local e procedimentos definidos conforme o gabarito da classe X . Agora estes procedimentos são específicos do objeto x e só eles têm acesso a sua memória local.

Uma classe é, portanto, uma entidade genérica representada pelo gabarito que a define, ou seja, o gabarito da classe é quem dá as características de seus objetos exemplares (instâncias). Podemos dizer ainda, por abuso de expressão, que os objetos herdam as características da classe.

3.3 SISTEMA DE SUPORTE

Um sistema de suporte é uma entidade que gerencia os objetos e realiza a comunicação na comunidade de objetos. Cabe a ele, a pedido, criar ou delir objetos. Sempre que o sistema de suporte recebe um pedido de criação, ele cria um objeto a imagem e semelhança do gabarito da classe solicitada no pedido e devolve uma mensagem para o solicitante dizendo que o atendeu. A partir deste instante o objeto criado já pode postar ou receber mensagens. Para suportar a comunicação entre objetos, o sistema de suporte faz o papel de correio. Coleta uma mensagem de um objeto remetente e se encarrega de entregá-la ao objeto destinatário.

Se o sistema de suporte recebe um pedido de deleção de um objeto e todas as condições para a deleção são verificadas, então o objeto será delido. E a partir do instante em que um objeto é delido ele deixa de existir para a comunidade de objetos.

O sistema de suporte, na literatura técnica, também é considerado como um objeto, a despeito de ele não ter um gabarito de uma classe que descreva suas características e nem ter sido criado por nenhum outro objeto.

Uma abrangência maior sobre este tema foge ao escopo do presente texto. Contudo, existem várias referências para o leitor interessado em se aprofundar mais no assunto. Uma formalização do conceito de objetos pode ser encontrada em Silva e Richter(1989). Outros artigos sobre o assunto são Cohen(1984), Cunnighan & Beck(1986), Hewitt(1973), Jonathan(1987), Matich(1988), Nguyen(1986), Rossi(1987), Rotenberg(1987), Strom(1986), Takahashi(1988) e Tokoro(1986).

4. COMBINAÇÃO DE EVIDÊNCIAS HIERARQUIZADAS

4.1 HIERARQUIA DE PROPOSIÇÕES

A figura 4.1 ilustra uma hierarquia de proposições do quadro de discernimento Θ igual a $\{\text{afif, Brizola, Collor, Covas, Freire, Lula, Maluf, Ullisses}\}$. Cada elemento $x \in \Theta$ representa a proposição: "x será eleito presidente da República". Os candidatos podem ser agrupados nas facções: Esquerda, Centro e Direita. Por sua vez as facções são reunidas na raiz com o nome Candidatos. A raiz representa todo o quadro de discernimento original Θ . Associado a cada nó x da árvore, há um quadro de discernimento Θ_x que é dado por:

$\Theta_x = \{x, \bar{x}\}$, $\forall x \in \{\text{Esquerda, Centro, Direita}\}$, significando:
"A facção x elege um candidato" ou
"A facção x não elege um candidato"

$\Theta_x = \{x, \bar{x}\}$, $\forall x \in \Theta$, isto "x será eleito" e "x não será eleito"

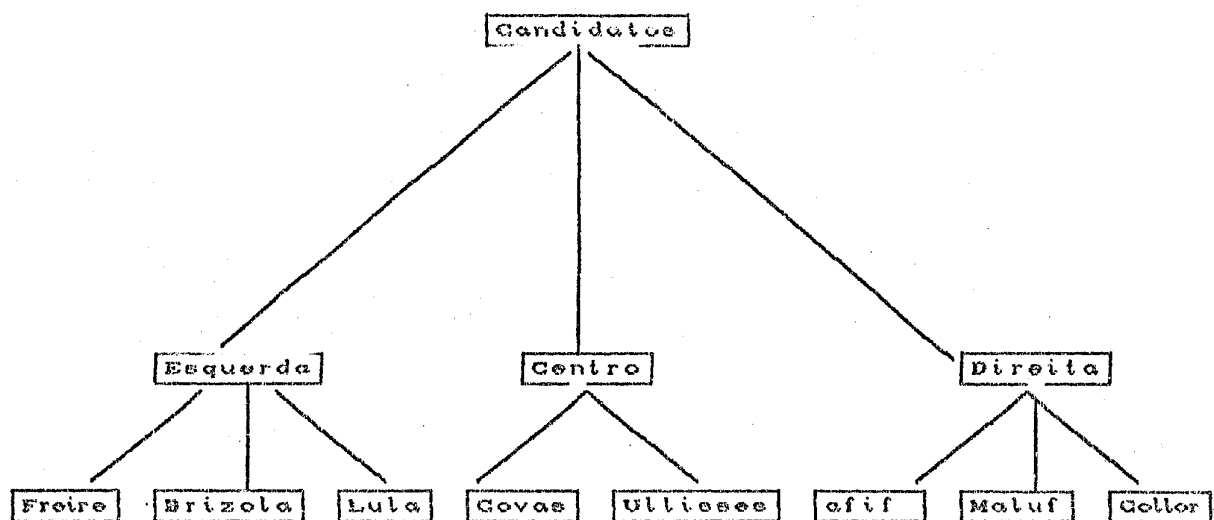


Figura 4.1 Hierarquia de hipóteses

O nó Esquerda é igual a $\{\text{Freire, Brizola, Lula}\}$, assim cada nó é a união de seus nós filhos.

Seja Θ um conjunto de proposições. Diz-se que uma árvore $A=(V,E)$ hierarquiza as proposições em Θ , se as seguintes condições são satisfeitas: A tem uma única raiz r ; cada nó $x \in V$ tem um conjunto de filhos F_x com $|F_x|=0$ ou $|F_x| \geq 2$; a raiz r representa a união de todas as proposições elementares, ou seja $r = \Theta$; cada nó interno, diferente da raiz r , representa a união de todos os seus

nós filhos; e cada nó folha representa uma proposição elementar de Θ , ou seja $x \in \Theta$.

Deste modo tem-se que a quantidade de nós folhas de \mathbb{A} é igual a cardinalidade de Θ . Usando o fato de cada nó interno ter ao menos dois filhos tem-se que $|V| < 2|\Theta|$.

Com esta definição de hierarquia de hipóteses não há perda de generalidade, já que está apenas eliminada a trivialidade de um nó ter um único filho, o que corresponderia a um refinamento identidade.

Nesse tipo de hierarquia a cada nó da árvore está associado um quadro de discernimento. Assim se $x \in V$, então $\Theta_x = (x, \bar{x})$ é o quadro de discernimento associado ao nó x . Deste modo, cada nó x em \mathbb{A} pode representar:

- Uma proposição do quadro de discernimento Θ_x associado ao nó x .
- Um subconjunto de proposições do quadro de discernimento Θ .
- A união dos subconjuntos de proposições de Θ representada pelos nós filhos de x .

Neste texto, nó x ou proposição x significará a proposição x do quadro Θ_x .

Cada quadro Θ_x é uma condensação de Θ , onde $\omega_x(x) = F_x$, $\omega_x(\bar{x}) = \bar{F}_x$ e ω_x é a função que dá o refinamento de Θ_x em Θ . O conjunto de todos os quadros de discernimento de \mathbb{A} forma uma família de quadros compatíveis, uma vez que todos eles têm Θ como um refinamento comum. Uma família de quadros compatíveis forma uma estrutura que permite induzir funções de crença de um quadro para qualquer outro quadro da família, via um refinamento comum aos quadros envolvidos na indução da crença, não necessariamente Θ .

Dado um nó x da árvore \mathbb{A} , um refinamento comum ao quadro Θ_x e aos quadros Θ_y dos filhos de x é o quadro de discernimento $\Omega_x = F_x \cup \bar{F}_x$. Por exemplo, o quadro $\Omega_{\text{Esquerda}} = \{\text{Freire, Brizola, Lula, Esquerda}\}$ é um refinamento comum aos quadros Θ_{Esquerda} , Θ_{Brizola} , Θ_{Freire} e Θ_{Lula} . Por este refinamento comum podem ser induzidas funções de crença de um dos quadros condensados para outro qualquer dos quadros condensados. Por exemplo, pode-se combinar as evidências discernidas pelos nós irmãos sobre o refinamento comum e induzir a função de crença resultante sobre o nó pai.

4.2 EVIDÊNCIAS

No problema proposto por Gordon e Shortliffe(1985), as evidências coletadas suportam crenças a favor ou contra as proposições representadas pelos nós da árvore A , que hierarquiza uma instância do problema. Este trabalho também aborda apenas este tipo de evidências.

Com frequência, neste texto, tem-se comentado sobre "Combinação de Evidências". Na verdade as evidências nunca são combinadas diretamente. O que a Soma Ortogonal combina são funções de crença suportadas por evidências. Se há um corpo de evidências ξ disponível, então cada peça de evidência de ξ dá origem a uma função de crença, as quais serão consideradas como pertencentes a um conjunto ζ . O que realmente compõe a entrada para o problema de combinações de evidências em um espaço de hipóteses hierarquizado é:

- Θ - Um quadro de discernimento, constituído por um conjunto de proposições exaustivas e exclusivas.
- $A=(V,E)$ - Uma árvore que hierarquiza as proposições de Θ .
- ζ - Um conjunto de funções de crença, onde cada função está na representação \mathcal{M} . Esta representação identifica o quadro de discernimento (uma condensação de Θ) sobre o qual a função está definida.

Deste modo, o processo de passagem da peça de evidência para função de crença que esta peça de evidência suporta não será considerado aqui. Contudo vale ressaltar que este processo é epistêmico, portanto varia de pessoa para pessoa. Na área de sistemas baseados em conhecimento é comum representar este processo de conversão por um conjunto de regras de produção, cujos antecedentes podem ser uma conjunção de possíveis manifestações da natureza (evidências) e o conseqüente a especificação da função de crença suportada por aquelas manifestações.

Sejam A_x uma subárvore de A com raiz $x=r$, e $\Theta_x=(x,\bar{x})$ o quadro de discernimento associado ao nó x de A . As funções de crença que são geradas a partir das evidências disponíveis podem ser classificadas em função de cada nó de A . Assim são definidos:

- ζ_x - Conjunto de funções de crença suportadas pelas

evidências discriminadas pelo nó x . Cada função de ζ_x é uma função de suporte simples (fss) a favor ou contra a proposição representada pelo nó x . Isto é, cada função ζ_x tem foco em x ou, alternativamente, em \bar{x} . A combinação de todas estas funções de crença resulta em uma função de crença dicotômica L_x com dicotomia (x, \bar{x}) . Esta função de crença é suportada por todo o corpo de evidências que gerou ζ_x .

- ζ_x^- - Conjunto de funções de suporte simples a favor ou contra proposições representadas pelos nós da subárvore A_x . Isto é,

$$\zeta_x^- = \bigcup_{t \in A_x} \zeta_t$$

- ζ_x^+ - Conjunto de todas as funções de suporte simples a favor ou contra proposições representadas por nós não pertencentes a subárvore A_x . Isto é,

$$\zeta_x^+ = \bigcup_{t \notin A_x} \zeta_t$$

- ζ - Conjunto de todas as funções de suporte simples dada como entrada do problema. Isto é, $\zeta = \zeta_x^- \cup \zeta_x^+, \forall x \in A$.

Finalmente, o problema é obter a combinação de todas as funções em ζ sobre cada quadro de discernimento $\Theta_x, \forall x \in A$, exceto $x=r$. Sabe-se que o processo de combinação de evidências é NP-Completo. Isto é, não é conhecido nenhum algoritmo para combinação de evidências com complexidade polinomial. Contudo, neste caso particular, as características do problema podem ser exploradas de modo a obter-se um algoritmo de complexidade linear.

4.3 FLUXO DE INFORMAÇÃO ENTRE OS NÓS

Considerando cada nó como um objeto independente, capaz de receber mensagens e enviar mensagens, conforme o paradigma de objetos, pode-se supor duas formas de reação dos nós: síncrona ou assíncrona.

Dentro da suposição sincronizada, em uma árvore de hierarquia, pode-se ver cada nó x de A recebendo e combinando todas as fss do seu conjunto ζ_x , e gerando a sua função de crença local L_x . Cada

nó folha f ao terminar o cálculo de L_f manda esta função de crença como uma mensagem $\lambda_f (=L_f)$ para o seu pai. Cada pai x , após receber a mensagem λ de todos os seus filhos, combina estas mensagens com a sua própria crença local L_x , gerando a mensagem λ_x . A seguir, envia esta mensagem para o seu próprio pai. O processo continua assim até que a raiz r de \mathbb{A} tenha recebido a mensagem λ de todos os seus filhos. Agora, a raiz r envia para cada filho x uma mensagem Π_x que é a combinação dos λ de todos os irmãos de x . Por sua vez, cada nó x , filho da raiz, envia uma mensagem Π para cada um de seus próprios filhos. Assim, cada filho f de x recebe uma mensagem Π_f , que é a combinação da mensagem Π_x , que x recebeu do seu pai mais a mensagem λ_x , que x enviou ao seu pai, descontada a mensagem λ_f , que o filho f enviou a x . O processo descendente continua. Cada nó, ao receber a mensagem Π do seu pai, também envia uma mensagem Π para cada um dos seus filhos, no mesmo esquema que os filhos da raiz enviaram mensagens aos seus filhos. Este processo de transmissão de mensagens de pai para filho só termina nos nós folhas. Quando então cada nó x de \mathbb{A} terá duas funções de crença λ_x e Π_x , as quais combinadas geram a crença C_x , sobre o quadro Θ_x . A crença C_x pode ser vista como sendo a crença induzida do resultado da combinação de todas as fss de ζ sobre o quadro discernimento Θ .

Dentro da suposição de reação assíncrona, pode-se ver cada nó x de \mathbb{A} , enviando uma mensagem λ ao pai e mensagens Π aos filhos a vista de cada fss do conjunto ζ_x , ou ainda a vista de cada mensagem λ ou Π que ele receba dos filhos ou do pai, respectivamente. Assim cada fss de ζ causa uma perturbação em todos os nós de \mathbb{A} , atualizando a crença C_x de forma mais dinâmica. A visão assíncrona será vista com mais detalhes na seção 6.

4.4 PROPRIEDADES ESTRUTURAIS

As seis primeiras propriedades a seguir descrevem importantes aspectos estruturais da hierarquia.

Seja Θ um quadro de discernimento cujas proposições possam ser estruturadas por uma árvore $\mathbb{A}=(\mathbb{W},\mathbb{F})$, e as evidências coletadas são tais que suportam crenças, diretamente, a favor ou contra as proposições representadas pelos nós de \mathbb{A} .

Propriedade 1. Para todo nó x de \mathbb{A} , exceto a raiz, existe um quadro $\Theta_x = (x, \bar{x})$ que é uma partição de Θ .

Demonstração: O quadro $\Theta_x = (x, \bar{x})$ tem o elemento x representando a união de todas as proposições folha da subárvore \mathbb{A}_x , e o complemento de x , então condensa todas as outras proposições de Θ que não são folhas de \mathbb{A}_x . Assim Θ_x é uma partição de Θ , pois $\forall \theta \in \Theta$, $\theta \in x$ ou $\theta \in \bar{x}$.

Propriedade 2. Sejam x um nó interno de \mathbb{A} , exceto a raiz, e F_x o conjunto dos nós filhos de x . Então $\Omega_x = F_x \cup (\bar{x})$ é um refinamento de $\Theta_x = (x, \bar{x})$.

Demonstração: Seja $\omega_x: 2^{\Theta_x} \longrightarrow 2^{\Omega_x}$, com $\omega_x((x)) = F_x$ e $\omega_x((\bar{x})) = (\bar{x})$ tem-se a função de refinamento pretendida.

Propriedade 3. Sejam x um nó interno de \mathbb{A} , e F_x o conjunto dos nós filhos de x . Então, para cada $y \in F_x$, $\Theta_y = (y, \bar{y})$ é uma condensação de $\Omega_x = F_x \cup (\bar{x})$.

Demonstração: Definindo $\omega_y: 2^{\Theta_y} \longrightarrow 2^{\Omega_x}$, com $\omega_y((y)) = (y) \subset F_x$ e $\omega_y((\bar{y})) = (F_x \setminus (y)) \cup (\bar{x})$, tem-se a condensação pretendida.

Propriedade 4. Seja x um nó interno de \mathbb{A} , exceto a raiz, Ω_x é uma condensação e uma partição de Θ .

Demonstração: Seja x um nó interno de \mathbb{A} , exceto a raiz, $\Omega_x = F_x \cup (\bar{x})$, e $\omega_x: 2^{\Omega_x} \longrightarrow 2^{\Theta}$. As propriedades da função ω_x asseguram que

$$\Theta = \bigcup_{z \in \Omega_x} \omega_x((z)), \text{ e}$$

$$\omega_x((z)) \cap \omega_x((z')) = \emptyset, \text{ se } z \neq z' \text{ e } z, z' \in \Omega_x$$

Logo Ω_x é uma condensação de Θ . E como, $\forall \theta \in \Theta$, existe apenas um $z \in \Omega_x$, tal que $\theta \in \omega_x((z))$, e z é uma proposição equivalente a $\omega_x((z))$, segue-se que Ω_x é equivalente a uma partição de Θ .

Propriedade 5. Se r é a raiz de \mathbb{A} , então F_r é uma condensação e uma partição de Θ .

Demonstração: Fazendo $r=x$ e $F_r = \Omega_x$, a prova é idêntica a demonstração da propriedade 4.

Propriedade 6. Sejam r a raiz de A , e F_r o conjunto dos nós filhos de r . Então, para todo $y \in F_r$, $\Theta_y = (y, \bar{y})$ é uma condensação de F_r .
 Demonstração: Seja $\omega_y : 2^{\Theta_y} \rightarrow 2^{F_r}$, um refinamento de Θ_y em F_r , onde F_r é o conjunto de filhos da raiz de A , e Θ_y é o quadro associado ao nó $y \in F_r$. Tomando $\omega_y((y)) = \{y\} \subset F_r$ e $\omega_y((\bar{y})) = F_r \setminus \{y\}$, $\forall y \in F_r$, tem-se a condensação pretendida.

4.5 CALCULO DE CRENÇAS

Nesta seção são apresentados os resultados que garantem a correção da combinação de evidências através do fluxo de mensagens entre os nós da hierarquia. Os lemas de 1 a 8 contêm relevantes relações entre as diversas funções de crença disponíveis na hierarquia. E, as proposições de 1 a 5 apresentam as fórmulas de cálculo necessárias a troca de mensagens entre os nós, para a obtenção das crenças combinadas.

Lema 1. Sejam λ e Π duas funções dicotômicas, sobre Ω , com dicotomia comum (x, \bar{x}) . A soma ortogonal, $\lambda \oplus \Pi$, é dada por

$$\lambda \oplus \Pi(x) = K \left[\lambda(x) \left(1 - \Pi(\bar{x}) \right) + \Pi(x) \left(1 - \lambda(x) - \lambda(\bar{x}) \right) \right],$$

$$\lambda \oplus \Pi(\bar{x}) = K \left[\lambda(\bar{x}) \left(1 - \Pi(x) \right) + \Pi(\bar{x}) \left(1 - \lambda(x) - \lambda(\bar{x}) \right) \right] e$$

$$\lambda \oplus \Pi(\Omega) = 1 - (\lambda \oplus \Pi(x) + \lambda \oplus \Pi(\bar{x}))$$

onde

$$K^{-1} = 1 - \lambda(x)\Pi(\bar{x}) - \lambda(\bar{x})\Pi(x)$$

Demonstração: Segue-se diretamente da regra de Dempster.

Lema 2. (Barnett) Seja Ω uma partição de Θ , e suponha que para cada $f \in \Omega$ exista uma função de crença dicotômica β_f , definida em Ω , com dicotomia (f, \bar{f}) . Onde $0 \leq \beta_f(x) < 1$, $\forall x \in (f, \bar{f})$. Assim, pode-se omitir o índice de $\beta_f(f)$ e $\beta_f(\bar{f})$ e escrever $\beta(f)$ e $\beta(\bar{f})$, respectivamente. Então a crença B definida por

$$B(y) = \left\{ \bigoplus_{f \in \Omega} \beta_f \right\} (y)$$

pode ser calculada usando

$$B(w) = K \left[\beta(w) \prod_{\substack{f \in \Omega \\ f \neq w}} (1 - \beta(f)) + \left(1 - \beta(w) - \beta(\bar{w}) \right) \prod_{\substack{f \in \Omega \\ f \neq w}} \beta(\bar{f}) \right], \quad \forall w \in \Omega \quad (4.1)$$

$$B(z) = \left(\bigoplus_{f \in \Omega} \beta_f \right) (z) \quad (4.2)$$

$$= K \left[\left[\prod_{f \in \Omega} (1 - \beta(f)) \right] \left[\sum_{f \in Z} \frac{\beta(f)}{1 - \beta(f)} \right] + \left[\prod_{f \in Z} \beta(\bar{f}) \right] \left[\prod_{f \in Z} (1 - \beta(f)) \right] - \prod_{f \in \Omega} \beta(\bar{f}) \right],$$

$\forall z \in \Omega, |z| \geq 2$

onde

$$K^{-1} = \left[\prod_{f \in \Omega} (1 - \beta(f)) \right] \left[1 + \sum_{f \in Z} \frac{\beta(f)}{1 - \beta(f)} \right] - \prod_{f \in \Omega} \beta(\bar{f}) \quad (4.3)$$

Demonstração: Veja Barnett(1981), ou Shafer e Logan(1987).

Lema 3. Sejam x um nó interno de \mathbb{A} , F_x o conjunto dos filhos de x , $\Omega_x = F_x \cup \{\bar{x}\}$, e $\Theta_x = (x, \bar{x})$. Se para cada $f \in F_x$ há uma função dicotômica β_f , induzida de Θ_x sobre Ω_x , com dicotomia (f, \bar{f}) , então

$$F_x = \left(\bigoplus_{f \in F_x} \beta_f \right)$$

e pode ser calculado como

$$F_x(x) = 1 - K \quad e$$

$$F_x(\bar{x}) = K \prod_{f \in F_x} \frac{\beta(\bar{f})}{1 - \beta(f)}$$

onde

$$K^{-1} = 1 + \sum_{f \in F_x} \frac{\beta(f)}{1 - \beta(f)}$$

Demonstração: Segue-se do lema 2. Pela propriedade 4, tem-se que Ω_x é uma partição de Θ , logo o lema 2 (de Barnett) se aplica. Considere que a função de β_x é vacuosa, daí se segue a demonstração. Onde $F(x)$ sai da fórmula (4.1) e $F(\bar{x})$ da fórmula (4.2), ambas com a simplificação do termo comum $\prod(1 - \beta(t))$, presente na fórmula (4.3). Com a simplificação de (4.3) em relação a $F(x)$ e $F(\bar{x})$ se obtém o termo de normalização K do presente lema.

Lema 4. Seja Ω uma partição de Θ . Se para cada $f \in \Omega$ houver uma função dicotômica β_f , sobre Ω , com dicotomia (f, \bar{f}) , então para todo $y \in \Omega$, tem-se:

$$\left[\prod_{\substack{f \in \Omega \\ f \neq y}} \beta_f \right] (y) = K \prod_{\substack{f \in \Omega \\ f \neq y}} \frac{\beta(\bar{f})}{1-\beta(f)}$$

$$\left[\prod_{\substack{f \in \Omega \\ f \neq y}} \beta_f \right] (\bar{y}) = 1 - K$$

onde

$$K^{-1} = 1 + \sum_{\substack{f \in \Omega \\ f \neq y}} \frac{\beta(f)}{1-\beta(f)}$$

Demonstração: Segue-se do Lema 2.

Lema 5. Sejam x um nó de \mathbb{A} , F_x o conjunto dos filhos de x , $\Omega_x = F_x \cup \{\bar{x}\}$, e $\Theta_y = (y, \bar{y})$, para todo $y \in F_x$. Se para cada $f \in F_x$ há uma função dicotômica λ_f , induzida de Θ_f sobre a partição Ω_x , com dicotomia (f, \bar{f}) , e uma função dicotômica LP_x , induzida de Θ_x sobre Ω_x , com dicotomia (\bar{x}, x) , então para cada $y \in F_x$ tem-se:

$$\Pi_y(y) = K \left[\prod_{\substack{f \in F_x \\ f \neq y}} \frac{\lambda(\bar{f})}{1-\lambda(f)} \right] \frac{LP(x)}{1-LP(\bar{x})} \quad e$$

$$\Pi_y(\bar{y}) = 1 - K$$

onde

$$K^{-1} = 1 + \sum_{f \in F_x} \frac{\lambda(f)}{1-\lambda(f)} + \frac{LP(\bar{x})}{1-LP(x)}$$

Demonstração: Segue-se do lema 2, dado que Ω_x é uma partição de Θ , pela propriedade 4.

Lema 6. Sejam x um nó interno de \mathbb{A} , exceto a raiz; L_x a função de crença local ao nó x , sobre Θ_x , com dicotomia (x, \bar{x}) . Se $L_x|_{\Omega_x}$ é uma função de crença sobre Ω_x , com dicotomia (F_x, \bar{x}) , e $L_x(z) = L_x|_{\Omega_x}(\omega(z))$, $\forall z \in \Theta_x$, então L_x e $L_x|_{\Omega_x}$ são funções de crença consistentes (induzidas uma da outra).

Demonstração: Segue-se da definição de funções de crença consistentes. Dado que Ω_x é um refinamento de Θ_x , pelo lema 2.

Lema 7. Sejam x um nó interno de \mathbb{A} , exceto a raiz, e λ_f uma função de crença dicotômica, sobre Θ_f , com dicotomia (f, \bar{f}) . Se $\lambda_f|_{\Omega_x}$ é uma função de crença dicotômica, sobre Ω_x , com dicotomia (f, \bar{f}) , e $\lambda_f(z) = \lambda_f|_{\Omega_x}(w(z))$, $\forall z \in \Theta_f$, então λ_f e $\lambda_f|_{\Omega_x}$ são funções de crenças consistentes, $\forall f \in \mathbb{F}_x$.

Demonstração: Segue-se da definição de funções de crenças consistentes. Dado que Ω_x é um refinamento de Θ_f pela propriedade 3.

Lema 8. Sejam r a raiz de \mathbb{A} , e λ_f uma função de crença dicotômica sobre Θ_f , com dicotomia (f, \bar{f}) . Se $\lambda_f|_{\mathbb{F}_r}$ é uma função de crença dicotômica, sobre \mathbb{F}_r , com dicotomia (f, \bar{f}) , e $\lambda_f(z) = \lambda_f|_{\mathbb{F}_r}(w(z))$, $\forall z \in \Theta_f$, então λ_f e $\lambda_f|_{\mathbb{F}_r}$ são funções de crenças consistentes, $\forall f \in \mathbb{F}_r$.

Demonstração: Segue-se da definição de funções de crenças consistentes. Dado que \mathbb{F}_r é um refinamento de Θ_f , pela propriedade 3, pois r aqui representa \emptyset e o complemento de \emptyset é vazio.

Pela propriedade 1, todo nó $x \in \mathbb{A}$, exceto a raiz, tem um quadro de discernimento Θ_x associado. Sabe-se que cada nó x tem uma função de crença local L_x , sobre Θ_x , que representa a combinação de todas as funções de crença da conjunto ζ_x . Mas para calcular a crença global C_x sobre cada nó x , exceto a raiz, é necessário combinar as crenças locais dos demais nós de \mathbb{A} com a crença local de x . Este trabalho pode ser feito usando as características da hierarquia de proposições e o algoritmo de Barnett(1981).

Conforme o item 4.3, λ_x representa a combinação das fss que têm algum nó da subárvore \mathbb{A}_x , ou o seu complemento, como foco. A função de crença λ_x está definida sobre Θ_x . Por sua vez, cada nó y filho de x ($y \in \mathbb{F}_x$) é a raiz de uma subárvore \mathbb{A}_y e, também, tem uma função de crença λ_y , sobre Θ_y . Mas pelas propriedades 2 e 3, Θ_x e os Θ_y , $\forall y \in \mathbb{F}_x$, têm um refinamento comum Ω_x . Assim, pelos lema 6 e 7, as funções de crença λ_y , $\forall y \in \mathbb{F}_x$, e L_x podem ser levadas para o quadro de discernimento Ω_x , gerando funções de crença $\lambda_y|_{\Omega_x}$, $\forall y \in \mathbb{F}_x$ e $L_x|_{\Omega_x}$. Estando estas funções sobre Ω_x , elas são combinadas e o resultado induzido sobre Θ_x para obter a função de crença λ_x . A combinação destas funções sobre Ω_x pode ser feita pelo algoritmo

de Barnett(1981), dado que Ω_x é uma partição de Θ , como assegura a propriedade 4. O cálculo de λ_x pode ser decomposto em dois componentes:

$$F_x|_{\Omega_x} = \left[\begin{array}{c} \oplus \lambda_t|_{\Omega_x} \\ \text{t} \in F_x \end{array} \right] \text{ e } L_x|_{\Omega_x}$$

os quais combinados produzem

$$\lambda_x = [(F_x|_{\Omega_x}) \oplus (L_x|_{\Omega_x})] |_{\Theta_x}$$

Alternativamente, pode-se induzir apenas os λ_y , $y \in F_x$, sobre Ω_x , obtendo as funções $\lambda_y|_{\Omega_x}$, as quais são combinadas conforme lema 2, e o resultado induz a função F_x , sobre Θ_x . Em seguida combina-se as funções F_x e L_x diretamente sobre Θ_x , conforme o lema 1, obtendo-se λ_x , já sobre Θ_x . As formulas abaixo sintetizam isto:

$$F_x = \left[\begin{array}{c} \oplus \lambda_t|_{\Omega_x} \\ \text{t} \in F_x \end{array} \right] |_{\Theta_x} \quad (4.4)$$

$$\lambda_x = F_x \oplus L_x$$

Contudo se x é um nó folha, então $\lambda_x = L_x$, pois o único nó da subárvore \mathbb{A}_x é o próprio x , e o quadro de discernimento Θ_x associado a x só discerne as evidências que suportam as fss de ζ_x . Logo $\lambda_x = L_x$, e $\zeta_x^- = \zeta_x$. Deste modo λ_x está definido para qualquer nó x de \mathbb{A} , exceto a raiz.

Não será feita nenhuma distinção das funções L_x , F_x e λ_x , sobre Θ_x , e as $L_x|_{\Omega_x}$, $F_x|_{\Omega_x}$ e $\lambda_x|_{\Omega_x}$, sobre Ω_x , respectivamente, dado que todas tem as suas proposições equivalentes, discriminadas em Ω_x , com o mesmo termo identificador.

Proposição 1. Sejam x um nó da árvore \mathbb{A} , exceto a raiz, ζ_x^- , o conjunto das fss suportadas por evidências discernidas pelos quadros de discernimento Θ_y , $y \in \mathbb{A}_x$, e λ_x , uma função de crença dicotômica, sobre Θ_x , definida como:

$$\lambda_x = \left[\begin{array}{c} \oplus \lambda_t \\ \text{t} \in F_x \end{array} \right] \oplus L_x, \text{ se } x \text{ é um nó interno, e}$$

$$\lambda_x = L_x, \text{ se } x \text{ é um nó externo}$$

então λ_x é a função de crença resultante da combinação das fss do conjunto ζ_x^- .

Demonstração. (a) Se x é um nó externo, folha, então as únicas fss sobre quadros da subárvore A_x são as da conjunto ζ_x , logo $\lambda_x = L_x$, e $\zeta_x^- = \zeta_x$. (b) Se x é um nó interno, exceto a raiz, então a função de crença λ_x é a soma ortogonal das crenças λ_y , $\forall y \in F_x$, mais a crença local L_x , do próprio x . Aqui há dois casos a considerar. (1) Todos os filhos de x são folhas: neste caso, cada $\lambda_y = L_y$ e portanto λ_x é a soma ortogonal das funções locais dos seus filhos mais a sua própria função de crença local L_x . Assim, λ_x é o resultado da combinação das fss a favor ou contra algum nó da subárvore A_x , que por definição é o conjunto ζ_x^- . (2) Os filhos de x são quaisquer nós: neste caso, cada filho y de x tem a função de crença λ_y como um resultado da combinação das fss de ζ_y^- ; e como λ_x combina os λ dos filhos e a sua crença local tem-se que ela resulta da combinação de todas as fss com foco em f ou \bar{f} , para todo f na subárvore A_x , e portanto no conjunto $\zeta_x^- = (\cup \zeta_y^-) \cup \zeta_x$.

Proposição 2. Se x é um nó interno, exceto a raiz, e λ_x é como na proposição 1, então λ_x é dada por

$$\lambda_x = F_x \oplus L_x,$$

onde

$$F_x = \left(\bigoplus_{f \in F_x} \lambda_f \right)$$

Demonstração: O cálculo de F_x é feito conforme o lema 3, e a soma $F_x \oplus L_x$ conforme lema 1.

Uma forma iterativa de calcular λ_x é começar calculando esta função, primeiro para os nós folhas, depois para os pais destas folhas, em seguida para os pais destes últimos, e assim sucessivamente, até obter a função λ_x para todos os nós, $x \in F_r$. Onde r é a raiz da árvore A . Deste modo a função

$$F_r = \left(\bigoplus_{x \in F_r} \lambda_x \right), \quad \text{sobre o quadro } F_r$$

representa a combinação de todas as fss de ζ . A crença global, C_x , sobre Θ_x , em cada nó $x \in F_r$ é $C_x(x) = F_r(x) \circ C_x(\bar{x}) = F_r(\bar{x})$, $\forall x \in F_r$. Mas é necessário achar, também, a crença global sobre os outros (nós) quadros de discernimentos da árvore A . isto pode ser feito calculando uma função de crença Π_x , para cada nó $x \in A$, exceto a

raiz, que combine todas as fss da conjunto ζ_x^+ .

As funções Π_x , $\forall x \in A$, são definidas recursivamente. As condições de contorno são definidas para os filhos da raiz. Se r é a raiz, pela propriedade 6, F_r é um refinamento comum de todos os Θ_x , $\forall x \in F_r$, e pelo lema 8, as funções de crença λ_x , sobre Θ_x , podem ser levadas para o quadro F_r , gerando as funções $\lambda_x|_{F_r}$, sobre o quadro F_r , $\forall x \in F_r$. Como pela propriedade 5, F_r é uma partição de Θ , pode-se usar o lema 2 para combinar as funções $\lambda_x|_{F_r}$. Este cálculo pode ser feito conforme Lema 4. Feito isto, passa-se a ter para cada x , filho da raiz, as crenças λ_x e Π_x . As quais combinadas geram a crença C_x . Isto é, $C_x = \lambda_x \oplus \Pi_x$. Após descer mais um nível na árvore, são calculadas as funções de crença Π para os filhos dos filhos da raiz: aqui, ao nível de cada nó x , com crença Π_x definida, é feito o cálculo da função $LP_x = L_x \oplus \Pi_x$, sobre Θ_x . Como Θ_x e Θ_y , $\forall y \in F_x$, tem Ω_x como um refinamento comum, pelas propriedades 2 e 3, Ω_x é uma partição de Θ , e pela propriedade 4, pode-se usar o lema de Barnett para combinar as funções $LP_x|_{\Omega_x}$ e $\lambda_y|_{\Omega_x}$, $\forall y \in F_x$, e assim obter as funções Π_y , $\forall y \in F_x$. Este cálculo é feito conforme lema 5. A partir dos filhos da raiz, vai-se descendo na árvore e calculando as funções Π_x em função de $\Pi_{PAI(x)}$, de $L_{PAI(x)}$ e dos λ dos irmãos de x , até chegar as folhas. A proposição 3 formaliza estas idéias.

Proposição 3. Sejam y um nó da árvore A , $y \neq r$, e ζ_y^+ o conjunto de fss sobre quadros Θ_z , $z \in A_y$. A função Π_y é definida como:

$$\Pi_y = \begin{pmatrix} \oplus \lambda_t \\ t \in F_r \\ t \neq y \end{pmatrix}, \quad \text{se } y \in F_r, \text{ onde } r \text{ é a raiz de } A. \quad (4.5)$$

e

$$\Pi_y = \Pi_x \oplus L_x \oplus \begin{pmatrix} \oplus \lambda_t \\ t \in F_x \\ t \neq y \end{pmatrix}, \quad \text{se } y \in F_x, \text{ onde } x \text{ é pai de } y \text{ e } x \text{ não é a raiz de } A. \quad (4.6)$$

Esta função de crença Π_y é a combinação das fss do conjunto ζ_y^+ .

Demonstração: (a) Seja r a raiz de A . Sabe-se que se $y \in F_r$, λ_y é a combinação das fss de ζ_y^- . Se Π_y , $\forall y \in F_r$, é dado pela fórmula (4.5), então todas as fss sobre os quadros de discernimentos fora da

subárvore A_y entram no cálculo de Π_y . Segue-se que Π_y é a combinação das fss do conjunto ζ_y^+ . (b) Se x é um nó interno de A , exceto a raiz, e $y \in F_x$, então Π_y é dado pela fórmula (4.6), a qual combina os λ dos irmãos de y , com crença local L_x , do pai de y , mais a crença Π_x . Pela soma ortogonal dos irmãos de y as fss de $\zeta_x^- \setminus (\zeta_x^- \cup \zeta_y^-)$ são combinadas. Pela parcela Π_x as fss de ζ_x^+ são combinadas e finalmente pela parcela L_x as fss de ζ_x^- são combinadas. Deste modo fazendo a união dos conjuntos $\zeta_x^- \setminus (\zeta_x^- \cup \zeta_y^-)$, ζ_x^+ e ζ_x^- vem $(\zeta_x^- \setminus \zeta_y^-) \cup \zeta_x^+ = \zeta_y^+$. Segue-se que Π_y é a combinação das fss do conjunto ζ_y^+ .

Em adição, a expressão (4.5) pode ser calculada conforme o lema 4, fazendo $F_r = \Omega$, e

$$\Pi_y(y) = \left[\begin{array}{c} \oplus_{t \in \Omega} \beta_t \\ t=y \end{array} \right] (y) \quad e$$

$$\Pi_y(\bar{y}) = \left[\begin{array}{c} \oplus_{t \in \Omega} \beta_t \\ t=y \end{array} \right] (\bar{y})$$

e a expressão (4.6) pode ser calculada conforme o lema 5, fazendo:

$$LP = L_x \oplus \Pi_x, \text{ definida sobre } \Omega_x, \\ \lambda_t(t) = \lambda(t) \quad e \quad \lambda_t(\bar{t}) = \lambda(\bar{t})$$

Uma vez que cada nó $x \in A$ já tenha calculado suas funções λ_x e Π_x , então a crença global $C_x = \lambda_x \oplus \Pi_x$ pode ser calculada. Essa função é dicotômica sobre Θ_x , e resulta da combinação das fss do conjunto ζ . A proposição 4 resume estas idéias.

Proposição 4. Sejam x um nó de A , exceto a raiz, e ζ o conjunto das fss dada como entrada; e $C_x = \lambda_x \oplus \Pi_x$ uma função de crença dicotômica sobre Θ_x , com dicotomia (x, \bar{x}) , então C_x é a combinação das fss do conjunto ζ .

Demonstração: Seja x um nó de A , exceto a raiz. Pela proposição 1, a função de crença λ_x é a combinação das fss do conjunto ζ_x^- ; e pela proposição 3, a função de crença Π_x é a combinação das fss do conjunto ζ_x^+ . Segue-se que $C_x = \lambda_x \oplus \Pi_x$ é a combinação das fss do conjunto $\zeta = \zeta_x^- \cup \zeta_x^+$.

Proposição 5. Sejam x um nó interno de \mathbb{A} , \mathbb{U}_x o conjunto dos filhos de x , $\Omega_x = \mathbb{F}_x \cup \{\bar{x}\}$, β_f uma função de crença dicotômica, sobre $\Theta_f = \{f, \bar{f}\}$, e induzida sobre Ω_x , $\forall f \in \mathbb{F}_x$, e F_x a função de crença sobre Θ_x , induzida pela soma ortogonal das funções β_f sobre Ω_x . Se após calcular F_x , surge uma nova evidência, discriminada por algum nó em uma subárvore \mathbb{A}_y , $y \in \mathbb{F}_x$, então a nova F'_x pode ser calculada como se segue:

(a) Recupera o fator de renormalização

$$K \leftarrow 1 - F_x(x)$$

(b) Recupera o produtório de $\beta(\bar{f})/(1-\beta(f))$, $\forall f \in \mathbb{F}_x$.

$$P \leftarrow F_x(\bar{x})/K$$

(c) Atualiza o fator de renormalização K

$$K'^{-1} \leftarrow K^{-1} - \beta(y)/(1-\beta(y)) + \beta'(y)/(1-\beta'(y))$$

onde β_y e β'_y são as funções de crença sobre Θ_y , induzida sobre Ω_x , antes e depois da nova evidência.

(d) Atualiza o produtório P

(i) Seja nz o número de funções β_f , $f \in \mathbb{F}_x$, com $\beta(\bar{f})=0$.

(ii) Se $\beta'(\bar{y})=0$, ou $nz > 1$ ou $\beta(\bar{y}) \neq 0$ e $nz=1$ então

$$P' \leftarrow 0$$

Senão

(iii) Se $nz=0$, então

$$P' \leftarrow P \cdot (1-\beta(y))/\beta(\bar{y}) \cdot \beta'(\bar{y})/(1-\beta'(y))$$

Senão

$$(iv) \quad P' \leftarrow \prod_{\substack{f \in \mathbb{F}_x \\ f \neq y}} (\beta(\bar{f})/(1-\beta(f))) \cdot (\beta'(\bar{y})/(1-\beta'(y)))$$

(e) Atualiza a função F_x

$$F'_x(x) \leftarrow 1 - K'$$

$$F'_x(x) \leftarrow K' \cdot P'$$

Demonstração: Os itens (a) e (b) seguem-se da manipulação aritmética das fórmulas do lema 3.

O item (c) é a subtração da parcela $\beta(y)/(1-\beta(y))$ que compunha K^{-1} e a adição da mesma parcela atualizada após a evidência surgida na subárvore \mathbb{A}_y , atualizando o valor de K .

O item (d) obtém o novo valor do produtório P . Ele é composto

de $|F_x|$ fatores. a idéia é substituir um dos fatores no produtório pelo seu novo valor atualizado: O subitem (i) define o número de fatores n_z com zeros no produtório. O subitem (ii) faz o novo produtório igual a zero se o fator substituído ($\beta'(\bar{y})=0$) é zero, ou se há mais de um fator com valor zero, ou ainda se apenas um fator é zero mas não é ele o fator substituído. O subitem (iii) atualiza o produtório apenas pela substituição de um dos fatores, dado que nenhum dos fatores é zero. Se as condições dos itens (ii) e (iii) são falsas, então o único com zero é o fator substituído, o que implica em recalcular o produtório com o fator substituído para ter este produtório diferente de zero. Isto é feito no item (iv).

o item (e) segue-se do lema 3.

Portanto a proposta da proposição 5 é constituída de manipulações aritméticas, face as condições especificadas dos fatores ou parcelas de um produto ou soma, respectivamente.

A proposição 5 encontra aplicação no cálculo das mensagens Π no algoritmo sequencial, proposto na seção 5, e na atualização das mensagens λ e Π no algoritmo paralelo proposto na seção 6.

5. ALGORITMO SEQUENCIAL

5.1 MOTIVAÇÃO

Um algoritmo sequencial para combinação de evidências em um espaço de hipóteses hierarquizado foi proposto por Shafer e Logan (1987). Esse algoritmo apresenta uma complexidade quadrática no fator de ramificação da árvore devido ao tempo de computação da "subroutine 5", descrita naquele artigo. O Objetivo aqui, é propor um algoritmo com complexidade linear no número de proposições de Θ , isto é, $O(|\Theta|)$.

5.2 SUPOSIÇÕES

Este algoritmo está baseado nas seguintes suposições:

(a) Todas as evidências já foram coletadas e combinadas ao nível local dos diversos nós da árvore, gerando as funções de crenças locais.

(b) Todas as funções de crenças L , locais a cada nó x de A , $x \neq r$, são dicotômicas com dicotomia (x, \bar{x}) , e valores $0 \leq L(y) < 1$, $\forall y \in (x, \bar{x})$.

5.3 NOTAÇÃO

Na descrição do algoritmo será usada a seguinte notação:

$A = (V, E)$ - Árvore que hierarquiza o conjunto de proposições de Θ .

F - Conjunto de nós filhos

F_x - Conjunto de nós filhos do nó x

$F(.)$ - Função dicotômica dada pela soma ortogonal das funções de crença λ dos filhos de x e calculada em x e \bar{x} .

Folha(x) - É verdadeiro se x é um nó folha.

$L(.)$ - Função dicotômica local a todo nó x , e calculada em x e \bar{x} , como um resultado da soma das fss da lista ζ_x .

$\lambda(.)$ - " " calculada em x e \bar{x} , representando a soma ortogonal de F_x e L_x , e enviada como mensagem ao pai de x .

$\Pi(.)$ - Função dicotômica calculada em x e \bar{x} , representa a soma das fss de ζ_x .

r - Raiz da árvore A .

5.4 OBJETIVO E RESUMO

O objetivo do algoritmo é calcular a crença em cada nó x da árvore, associado ao qual há um quadro de discernimento Θ_x , sobre o qual a crença está definida. A idéia é combinar todas as funções de crença da lista ζ ao nível de cada quadro Θ_x , para cada nó x da árvore.

O algoritmo tem dois procedimentos principais Calcula_λ e Calcula_Π , descritos a seguir.

Calcula $_\lambda$: Calcula a mensagem λ que cada nó de \mathbb{A} envia ao seu pai. Usando um caminhamento em profundidade e em pós-ordem, cada nó x de \mathbb{A} é visitado. Cada visita consiste em calcular a mensagem λ em função das mensagens λ dos filhos de x e da crença local L de x . Se x é um nó folha então $\lambda=L$. Se x é um nó interno, também, é feita uma contagem da quantidade dos filhos y de x que têm $\lambda(\bar{y})=0$. O valor apurado é colocado no vetor $\text{nz}(x)$.

Calcula $_\Pi$: Usando um caminhamento em largura, todos os nós da árvore são visitados. A visita consiste em calcular a crença Π que cada nó interno manda ao seu filho e a crença global C , resultante da combinação de λ e Π . O processo termina quando todos os nós forem visitados.

5.5 ENTRADA E SAIDA

A entrada para o algoritmo se constitui da árvore $\mathbb{A} = (V, E)$, onde cada nó x de \mathbb{A} , exceto a raiz, tem a seguinte estrutura:

- Crença local: os dois valores $L(x)$ e $L(\bar{x})$;
- Soma dos λ dos filhos: os dois valores $F(x)$ e $F(\bar{x})$;
- Mensagem enviada ao pai: os dois valores $\lambda(x)$ e $\lambda(\bar{x})$;
- Mensagem recebida do pai: os dois valores $\Pi(x)$ e $\Pi(\bar{x})$;
- Crença final: os dois valores $C(x)$ e $C(\bar{x})$; e,
- Conjunto de endereços dos filhos: F_x .

O nó raiz de \mathbb{A} só tem em sua estrutura o conjunto de endereços dos seus filhos, F_r . A árvore de entrada \mathbb{A} é inicializada com todos os dados de cada nó x zerados, exceto os valores da crença local L_x e o conjunto de endereços F_x . A crença local L_x é o resultado da combinação das fss do conjunto ζ_x . O algoritmo supõe que estas combinações, para obtenção dos dois valores $L(x)$ e $L(\bar{x})$ de L_x , já tenham sido feitas.

Como saída o algoritmo define os valores para as demais funções de crenças de cada nó da árvore A , em função das crenças locais: $L(x)$ e $L(\bar{x})$. Os cálculos destes elementos são feitos conforme descrito na seção 4.

5.6 PSEUDO-CÓDIGO

Neste item é descrito o algoritmo em uma linguagem de nível bem alto, contudo o custo de cada operação é estimável o que permite a determinação da ordem de complexidade.

Início do Pseudo-Código ;

F : Fila ; (* F é global *)

Procedimento Calcula_ λ (x :nó) ;

Se folha(x), então

$\lambda(x) \leftarrow L(x)$;

$\lambda(\bar{x}) \leftarrow -L(\bar{x})$

Senão

$nz(x) \leftarrow 0$; (* conta qtos filhos de x tem $\lambda(\bar{y})=0$ *)

Para todo $y \in F_x$, faça

Calcula_ λ (y) ;

Se $\lambda(\bar{y}) = 0$ então $nz(x) \leftarrow nz(x) + 1$

FimPara ;

Se $x \neq r$, então

$$K_x^{-1} \leftarrow 1 + \sum_{t \in F_x} \frac{\lambda(t)}{1-\lambda(t)} ;$$

$$P_x \leftarrow \prod_{t \in F_x} \frac{\lambda(\bar{t})}{1-\lambda(t)} ;$$

$$F(x) \leftarrow 1 - K_x ; F(\bar{x}) \leftarrow K_x \cdot P_x ;$$

$$K^{-1} \leftarrow 1 - L(\bar{x})F(x) - L(x)F(\bar{x}) ;$$

$$\lambda(x) \leftarrow K \left[L(x) \left(1 - F(\bar{x}) \right) + F(x) \left(1 - L(x) - L(\bar{x}) \right) \right] ;$$

$$\lambda(\bar{x}) \leftarrow K \left[L(\bar{x}) \left(1 - F(x) \right) + F(\bar{x}) \left(1 - L(x) - L(\bar{x}) \right) \right]$$

FimSe ;

FimSe ;

FimProcedimento Calcula_ λ ;

Procedimento Calcula_Π ;

Enquanto $F \neq \emptyset$, faça

$x \leftarrow \text{Primeiro}(F)$;

Se não Folha(x), então (* Expande o nó x *)

Se $x \neq r$, então

(* combina a crença Local com a mensagem Π do Pai *)

$$K_x^{-1} \leftarrow 1 - L(x)\Pi(\bar{x}) - L(\bar{x})\Pi(x)$$

$$LP(x) \leftarrow K \left[L(x) \left(1 - \Pi(\bar{x}) \right) + \Pi(x) \left(1 - L(x) - L(\bar{x}) \right) \right]$$

$$LP(\bar{x}) \leftarrow K \left[L(\bar{x}) \left(1 - \Pi(x) \right) + \Pi(\bar{x}) \left(1 - L(x) - L(\bar{x}) \right) \right]$$

(* Recupera o conflito da influência dos filhos *)

$$K_x \leftarrow 1 - F(x)$$

$$K_x^{-1} \leftarrow K_x^{-1} + \frac{LP(\bar{x})}{1 - LP(\bar{x})}$$

(* recupera o produtório *)

$$P_x \leftarrow \frac{F(\bar{x}) \cdot LP(x)}{K_x \cdot 1 - LP(\bar{x})}$$

(* Verifica se a crença LP(x) é zero *)

Se $LP(x) = 0$, então $nz(x) \leftarrow nz(x) + 1$ FimSe ;

Senão (* $x = r$ *)

$$K_x^{-1} \leftarrow 1 + \sum_{t \in F_x} \frac{\lambda(t)}{1 - \lambda(t)} ;$$

$$P_x \leftarrow \prod_{t \in F_x} \frac{\lambda(\bar{t})}{1 - \lambda(t)} ;$$

FimSe ;

(* Calcula o Π de cada filho de x *)

Para todo $z \in F_x$, faça (* Visita todos os filhos de x *)

$$K_x^{-1} \leftarrow K_x^{-1} - \frac{\lambda(z)}{1 - \lambda(z)} ;$$

Se $nz(x) = 0$, então

$$P_x \leftarrow P_x \cdot \frac{1 - \lambda(z)}{\lambda(\bar{z})}$$

Senão

Se $\lambda(\bar{z})=0$ e $nz(x)=1$, então

$$P \leftarrow \prod_{\substack{t \in F \\ t \neq z^x}} \lambda(t) \quad ;$$

Se $x \neq r$, então $P \leftarrow P \frac{LP(x)}{1-LP(\bar{x})}$ FimSe ;

Senão

$$P \leftarrow 0 \quad ;$$

FimSe (* Se $\lambda(\bar{z})=0$ e $nz(x)=1$ *)

FimSe ; (* Se $nz(x) = 0$ *)

(* Cálculo da influência dos irmãos sobre z *)

(* Cálculo da mensagem que pai x manda para z *)

$$\Pi(z) \leftarrow K.P \quad ;$$

$$\Pi(\bar{z}) \leftarrow 1 - K \quad ;$$

(* Cálculo da crença global *)

$$K^{-1} \leftarrow 1 - \lambda(z)\Pi(\bar{z}) - \lambda(\bar{z})\Pi(z) \quad ;$$

$$C(z) \leftarrow K \left[\lambda(z) \left(1 - \Pi(\bar{z}) \right) + \Pi(z) \left(1 - \lambda(z) - \lambda(\bar{z}) \right) \right] \quad ;$$

$$C(\bar{z}) \leftarrow K \left[\lambda(\bar{z}) \left(1 - \Pi(z) \right) + \Pi(\bar{z}) \left(1 - \lambda(z) - \lambda(\bar{z}) \right) \right] \quad ;$$

$$F \leftarrow \text{Enfileira}(z, F) \quad ;$$

FimPara (* todo $z \in F$ *)

FimSe (* Se não Folha(x) *)

FimEnquanto (* $F \neq \phi$ *)

FimProcedimento ; (* Calcula Π *)

(* ::::::::::: Corpo Principal ::::::::::: *)

Calcula_λ(r) ; (* Calcula as mensagens λ *)

Esvazia(F) ; (* torna a fila Vazia *)

Enfileira(r, F) ; (* Põe a raiz r de A na fila *)

Calcula_Π ; (* Calcula Π e a crença final *)

FimPseudo-Código.

5.7 ANÁLISE DE COMPLEXIDADE

O tempo de computação gasto para combinar todas as crenças locais ao nível de cada nó da árvore A é igual a soma dos esforços dos procedimentos Calcula_λ e calcula_π . A seguir é analisada a complexidade em cada um destes procedimentos.

Calcula_λ é um Procedimento recursivo. Para se calcular a função λ_x de um dado nó x é necessário calcular primeiro as funções λ dos nós filhos de x . Assim Calcula_λ faz um caminhamento em pós-ordem na árvore A , e o trabalho de cálculo se encerra quando todos os filhos da raiz r de A tiverem suas funções λ definidas. O procedimento Calcula_λ também conta o número de filhos y de cada nó $x \in A$ que tem a função $\lambda(y) = 0$. O resultado desta contagem fica armazenado no vetor $\text{nz}(x)$. O esforço de cálculo para cada tipo de nó é dado por

- nó folha
 - . 2 atribuições.
- nó interno x
 - . 7 atribuições;
 - . $12 + 4|F_x|$ Somas; e,
 - . $10 + 3|F_x|$ produtos.

Seja $c = \max(\text{atribuição}, \text{soma}, \text{produto})$, a constante que representa o maior esforço dentre as operações citadas. Assim o esforço em cada nó folha é no máximo $2c$, e em cada nó interno é no máximo $(29 + 7|F_x|)c$. Na árvore existem $|\Theta|$ nós folhas e $|(V \setminus \Theta)|$ nós internos, incluída a raiz. Pela definição da árvore de hierarquia tem-se que $|V| \leq 2|\Theta|$. Portanto o esforço do procedimento Calcula_λ é menor que

$$2c|\Theta| + \sum_{x \in (V \setminus \Theta)} (29 + 7|F_x|)c = 45c|\Theta| \quad (5.1)$$

Calcula_π é um procedimento iterativo que faz um caminhamento em largura na árvore de hierarquia A . Ele faz uso de um fila F para armazenar os nós visitados e ainda não expandidos. A expansão de um nó x de A consiste em visitar todos os seus filhos. Em cada visita de um nó z filho de x , é calculada a mensagem Π_z , de x para z , e a

crença global C_z . O tempo de computação total do procedimento Calcula_Π é dado pelo tempo de expansão dos nós mais o tempo de visita dos nós. O tempo de expansão de cada nó interno é dado por

- Nó interno x , não raiz
- . 7 atribuições;
 - . $14 + |F_x|$ somas; e,
 - . $14 + 2|F_x|$ produtos.

- Nó raiz
- . 3 atribuições;
 - . $3|F_r|$ somas; e,
 - . $3|F_r|$ Produtos.

Considerando a constante c como na análise de Calcula_λ, o esforço máximo para expansão dos nós internos, é dado por

$$\sum_{\substack{x \in (V \setminus \Theta) \\ x \neq r}} (35 + 3|F_x|)c + (3 + 6|F_r|)c = 41|\Theta|c + 3|F_x|c + 3c \leq 44|\Theta|c$$

O tempo de visita de cada nó x de A é dado por

- . 7 atribuições;
- . 14 somas; e,
- . 14 produtos.

como todos os $|V|$ nós de A , exceto a raiz, são visitados, tem-se que o tempo total de visitação é menor que

$$35|V|c = 70|\Theta|c$$

Agora, somando o tempo de expansão mais o tempo de visitação dos nós de A , tem-se o tempo total do procedimento Calcula_Π que é menor que

$$(44|\Theta| + 70|\Theta|)c = 114|\Theta|c \quad (5.2)$$

Deve-se observar que o tempo gasto para recuperar o produtório P , quando a condição " $\lambda(\bar{z})=0$ e $nz(x)=1$ " dentro do comando "Para todo $z \in F_x$, faça " é verdadeira, foi considerado como um tempo de exploração do nó x . Pois esta condição é potencialmente verdadeira para, exatamente, um único filho de x . Isto é, o recálculo de P é feito, no pior caso, uma vez para cada nó interno x de A .

Somando o pior tempo gasto pelos procedimentos Calcula_λ e Calcula_Π tem-se o pior tempo total gasto pelo algoritmo para o problema de combinar evidências em um espaço de proposições hierarquizado. Somando (5.1) e (5.2) tem-se

$$(45|\Theta| + 114|\Theta|)c = 159|\Theta|c$$

Logo a complexidade de tempo é linear no tamanho de Θ , isto é, $T(|\Theta|) = O(|\Theta|)$.

A complexidade de espaço pode ser dada em termos das variáveis alocada em cada nó, do tamanho máximo da pilha de recursão usada no procedimento Calcula_λ e no tamanho da fila usada pelo procedimento Calcula_Π , das variáveis auxiliares envolvidas. A seguir é dado um resumo disto:

- Variáveis por nó

Em cada nó x de \mathbb{A} , é necessário armazenar os valores das funções: L , F , Π e λ . Cada uma delas tem dois valores de interesse. Logo cada nó gasta oito posições de valores reais. E como temos menos que $2|\Theta|$ nós na árvore, então a complexidade de espaço é $O(|\Theta|)$.

- Pilha de Recursão de Calcula_λ

A Pilha de recursão pode chegar a uma profundidade máxima de $\log_2 |\Theta|$. O contexto salvo por cada chamada é limitado por uma constante c . Logo a complexidade de espaço é $O(\log_2 |\Theta|)$.

- Fila de Calcula_Π

A fila armazena apenas os nós visitados e ainda não explorados. No pior caso esta fila chega a ter $|V|-1$ elementos. Portanto sua complexidade de espaço é $O(|\Theta|)$.

- Variáveis auxiliares

O procedimento Calcula_λ usa o vetor nz com $|V|$ componentes, e tanto Calcula_λ quanto Calcula_Π usam outras variáveis auxiliares em número irrelevante. Deste modo este tipo de variável tem complexidade de espaço $O(|\Theta|)$.

Pelo resumo se conclui que o espaço gasto para combinar evidências em um espaço de hipóteses hierarquizado é linear na quantidade de hipótese, isto é $S(|\Theta|) = O(|\Theta|)$.

6 ALGORITMO PARALELO

6.1 EXPLICITANDO O PARALELISMO

Em uma hierarquia de proposições, cada nó $x \in A$, exceto a raiz, tem sua crença global dada por $C_x = \lambda_x \oplus \Pi_x$. Isto é, a função de crença λ_x , resultante da combinação das fss do conjunto ζ_x^- , combinada com a função de crença Π_x , resultante da combinação das fss do conjunto ζ_x^+ . Conforme a seção 4, $\lambda_x (= L_x \oplus F_x)$ é a soma das funções de crença L_x (combinação das fss de ζ_x^-) e F_x (combinação das crenças λ dos filhos de x). Assim a função de crença C_x pode ser alterada nas seguintes situações:

(a) L_x é alterada:

Isto é possível quando uma fss do conjunto ζ_x^- é informada ao sistema e combinada com a função L_x produzindo L'_x .

(b) F_x é alterada

Isto é possível quando uma nova fss sobre um dos quadros de discernimento Θ_y , $y \in A_x \setminus \{x\}$, se torna disponível e é combinada com L_y , gerando uma propagação que altera F_x .

(c) Π_x é alterada

Isto é possível quando uma nova fss sobre um dos quadros de discernimento Θ_z , $z \in A_x$, se torna disponível e é combinada com a função de crença local L_z gerando uma propagação pela árvore que provoca a alteração de Π_x .

(d) Uma combinação qualquer das situações anteriores.

Pode-se considerar cada nó na árvore como um objeto semi-autônomo, com capacidade de receber mensagens, processar as informações pertinentes a estas mensagens e enviar mensagens a seus filhos e/ou a seu pai. Este enfoque é possível dado que interessa a cada nó x apenas a última mensagem que veio de seu pai (a última Π_x , substitui a anterior), bem como a última λ_y , que veio do filho $y \in F_x$, substitui a anterior. O único componente da crença global C_x , da mensagem λ_x , e das mensagens Π_y , $\forall y \in F_x$, que é cumulativo é a crença local L_x . Todos os demais componentes são recalculados a cada situação descrita por (b), (c) ou ambas simultaneamente, ao nível de cada nó x . A estrutura em árvore impede ressonâncias. Se uma nova fss se torna disponível em uma subárvore A_x , então cabe ao nó x enviar uma mensagem λ_x ao seu

pai, o qual nada manda de volta por conta desta mensagem. Isto impede a ressonância. Contudo, o pai de x dá conhecimento do impacto dessa fss para os irmãos de x e para o avô de x . Assim a fss que impactou a subárvore A_x se propaga na árvore, e todos os quadros de discernimento da árvore A , direta ou indiretamente, sofrem seu impacto.

6.2 PARADIGMA DE OBJETOS

O projeto de algoritmos usando o paradigma de orientação a objetos leva a ver recursos como objetos, e definir as operações realizadas sobre estes objetos de modo encapsulado. A representação dos recursos e suas operações ficam restritas a um módulo específico para cada recurso visto como um objeto. Qualquer referência a um objeto deve ser feita através de uma das operações definidas no módulo daquele objeto. As características mais importante deste paradigma, que o difere de tipo abstrato de dados, são os conceitos de CLASSIFICAÇÃO e HERANÇA. No caso de hierarquia de proposições, temos três classes de objetos: Os nós folhas, os nós internos, exceto a raiz, e a raiz. Temos em adição, para manter homogeneidade, uma classe de usuários. Assim Podemos definir um gabarito de dados e as operações para cada uma destas classes de objetos. E os objetos passam a ser apenas instâncias destas classes. Um objeto, ao ser criado, herda o gabarito de dados e procedimentos especificados em sua classe. Para mais detalhes sobre o paradigma de orientação a objetos veja Silva e Richter (1989), o qual apresenta uma formalização desse paradigma e também uma longa lista de referências sobre o tema.

6.3 DADOS E OPERAÇÕES POR CLASSE DE NÓS

Conforme secção 6.2, os objetos (nós) são exemplares (ou instâncias) de uma das classes de objetos. Definido o gabarito de dados e as operações destas classes; e um sistema de suporte que crie instâncias de objetos, receba mensagens de objetos e entregue mensagens a objetos; então todo o algoritmo estará especificado. Já que cada objeto é independente de outro objeto, e a única de forma de interação entre eles é através de mensagens. A cada objeto pode estar associado um processador, e assim vários

objetos podem estar realizando operações simultaneamente. A seguir é definido um gabarito para cada uma das classes acima.

6.3.1 Nó FOLHA

Id. nó: x

Id. pai: p

Dados: $L(x), L(\bar{x}); C(x), C(\bar{x}); \Pi(x), \Pi(\bar{x});$ Conforme notação da secção 6.1.

Operações:

(a) Inicialização

$$\left[L(x), L(\bar{x}); C(x), C(\bar{x}); \Pi(x), \Pi(\bar{x}) \right] \leftarrow 6 \times 0 ;$$

(b) CrençaGlobal [K]

$$K^{-1} \leftarrow 1 - L(x)\Pi(\bar{x}) - L(\bar{x})\Pi(x) ;$$

$$C(x) \leftarrow K \left[L(x) \left[1 - \Pi(\bar{x}) \right] + \Pi(x) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

$$C(\bar{x}) \leftarrow K \left[L(\bar{x}) \left[1 - \Pi(x) \right] + \Pi(\bar{x}) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

(c) QualCrença msg.: - ; dest.: x ; orig.: y .

Entregar msg: $C(x), C(\bar{x});$ dest.: y ; orig.: x .

(d) NovaEvidência msg.: $E(x), E(\bar{x})$; dest.: x ; orig.: usuário

[l, l', K] (* Variáveis temporárias *)

$$K^{-1} \leftarrow 1 - L(x)E(\bar{x}) - L(\bar{x})E(x) ;$$

$$l \leftarrow K \left[L(x) \left[1 - E(\bar{x}) \right] + E(x) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

$$l' \leftarrow K \left[L(\bar{x}) \left[1 - E(x) \right] + E(\bar{x}) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

$$L(x) \leftarrow l ;$$

$$L(\bar{x}) \leftarrow l' ;$$

(* Filho x (folha) pede ao pai z receber sua mensagem L *)

z MsgdeFilho msg: $L(x), L(\bar{x})$; dest.: z ; orig.: x ;

Minha CrençaGlobal ; (* Atualiza a crença global *)

(e) MsgdoPai msg.: $P(x)$, $P(\bar{x})$; dest.: x ; orig.: z

$$\Pi(x) \leftarrow P(x) \quad ;$$

$$\Pi(\bar{x}) \leftarrow P(\bar{x}) \quad ;$$

Minha CrençaGlobal ; (* Atualiza a crença global *)

6.3.2 Nó RAIZ

Id. nó: r

Lista de Filhos: F_r .

Dados: $(\lambda(x), \lambda(\bar{x})) | \forall x \in F_r$, K_r , P_r , nz ;

Operações:

(a) Inicialização

$$(\lambda(x), \lambda(\bar{x})) | \forall x \in F_r \leftarrow (0, 0) | \forall x \in F_r ;$$

$$K_r^{-1} \leftarrow 1 ; \quad P_r \leftarrow 0 ; \quad nz \leftarrow |F_r| ;$$

(b) MsgdeFilho msg: $f(x)$, $f(\bar{x})$; dest.: r ; orig.: x

$[K, P, y, t]$ (* variáveis temporárias *)

(* atualiza o conflito *)

$$K_r^{-1} \leftarrow K_r^{-1} - \frac{\lambda(x)}{1-\lambda(x)} + \frac{f(x)}{1-f(x)}$$

(* atualiza o produtório *)

Se $nz = 0$, então

$$P_r \leftarrow P_r \cdot \frac{1-\lambda(x)}{\lambda(\bar{x})} \cdot \frac{f(\bar{x})}{1-f(x)}$$

Senão

Se $nz=1$, $\lambda(\bar{x})=0$ e $f(x) > 0$, então

$$P_r \leftarrow \prod_{\substack{t \in F_r \\ t \neq y}} \frac{\lambda(\bar{t})}{1-\lambda(\bar{t})} \cdot \frac{f(\bar{t})}{1-f(\bar{t})} ;$$

Senão

$$P_r \leftarrow 0$$

FimSe

FimSe ;

(* Atualiza os zeros ainda restantes *)

Se $\lambda(\bar{x}) = 0$ e $f(\bar{x}) > 0$, então

$$nz \leftarrow nz - 1$$

Senão

Se $\lambda(\bar{x}) > 0$ e $f(x) = 0$, então

$$nz \leftarrow nz + 1$$

FimSe

FimSe ;
 (* Atualiza os λ *)
 $\lambda(x) \leftarrow f(x)$;
 $\lambda(\bar{x}) \leftarrow f(\bar{x})$;
 (* Calcular o Π que a raiz manda para cada filho *)
 Para todo $y \in F \setminus \{x\}$, faça

$$K^{-1} \leftarrow K_r^{-1} - \frac{\lambda(y)}{1-\lambda(y)} ;$$

Se $nz=0$, então

$$P \leftarrow P_r \cdot \frac{1-\lambda(y)}{\lambda(\bar{y})}$$

Senão

Se $nz=1$ e $\lambda(\bar{y}) = 0$, então

$$P \leftarrow \prod_{\substack{t \in F \\ t \neq y}} \frac{\lambda(t)}{1-\lambda(t)}$$

Senão

$$P \leftarrow 0$$

FimSe

FimSe ;

$$\Pi(y) \leftarrow K \cdot P ;$$

$$\Pi(\bar{y}) \leftarrow 1 - K ;$$

(* O Pai r pede que filho y receba a mensagem Π *)
 $y \text{ MsgdoPai } \text{msg: } \Pi(y), \Pi(\bar{y}) ; \text{dest.: } y ; \text{orig.: } r ;$

FimPara ;

6.3.3 Nó INTERNO (* exceto a raiz *)

Id. do pai: p ;

Id. nó: x ;

Lista de Filhos: F_x .

Dados: $(\lambda(y), \lambda(\bar{y})) \mid \forall y \in F_x$, $L(x)$, $L(\bar{x})$, $F(x)$, $F(\bar{x})$,
 $\lambda(x)$, $\lambda(\bar{x})$, $\Pi(x)$, $\Pi(\bar{x})$, $LP(x)$, $LP(\bar{x})$, nz ;

Operações:

(a) Inicialização

$$\{ (\lambda(y), \lambda(\bar{y})) \mid \forall y \in F_x \} \leftarrow \{ (0,0) \mid \forall y \in F_x \} ;$$

$$\{ L(x), L(\bar{x}); C(x), C(\bar{x}); F(x), F(\bar{x}); \Pi(x), \Pi(\bar{x}); LP(x), LP(\bar{x}) \} \leftarrow 8 \times 0 ;$$

$$nz \leftarrow |F_x| ;$$

(b) Mãe de Filho mãe: $f(y)$, $f(\bar{y})$; dest.: x ; orig.: y
 $[K_x, P_x, y, t]$ (* variáveis temporárias *)

(* Recupera o fator de renormalização da influência dos filhos *)

$$K_x \leftarrow 1 - F(x)$$

(* recupera o produtório *)

$$P_x \leftarrow \frac{F(\bar{x})}{K_x}$$

(* atualiza o fator de renormalização *)

$$K_x^{-1} \leftarrow K_x^{-1} - \frac{\lambda(y)}{1-\lambda(y)} + \frac{f(y)}{1-f(y)}$$

(* atualiza o produtório *)

Se $nz=0$, então

$$P_x \leftarrow P_x \cdot \frac{1-\lambda(y)}{\lambda(\bar{y})} \cdot \frac{f(\bar{y})}{1-f(y)}$$

Senão

Se $nz=1$, $\lambda(\bar{y})=0$, $f(\bar{y})=0$, então

$$P_x \leftarrow \prod_{\substack{t \in F \\ t \neq y^x}} \frac{\lambda(t)}{1-\lambda(t)} \cdot \frac{f(\bar{y})}{1-f(y)}$$

Senão

$$P_x \leftarrow 0$$

FimSe

FimSe ;

(* Atualiza os zeros ainda restantes *)

Se $\lambda(\bar{y}) = 0$ e $f(y) > 0$, então

$$nz \leftarrow nz - 1$$

Senão

Se $\lambda(\bar{y}) > 0$ e $f(\bar{y}) = 0$, então

$$nz \leftarrow nz + 1$$

FimSe

FimSe ;

(* atualiza a crença do filho y *)

$$\lambda(y) \leftarrow f(y) ;$$

$$\lambda(\bar{y}) \leftarrow f(\bar{y}) ;$$

(* atualiza a crença combinada dos filhos *)

$$F(x) \leftarrow 1 - K_x ;$$

$$F(\bar{x}) \leftarrow K_x \cdot P_x ;$$

Minha MandaMsgFilhos msg.:y ;
 Minha MandaMsgPai ;
 Minha CrençaGlobal ;

(c) MandaMsgFilhos msg: y;

$[K_x, P_x, t, z, K, P, nx]$

$K_x \leftarrow 1 - F(x)$

$K_x^{-1} \leftarrow K_x^{-1} + \frac{LP(\bar{x})}{1-LP(\bar{x})}$

(* recupera o produtório *)

$P_x \leftarrow \frac{F(\bar{x})}{K_x} \cdot \frac{LP(x)}{1-LP(\bar{x})}$

(* Verifica quantos \bar{z} , $z \in F_x$, tem $\lambda(\bar{z})=0$ *)

$nz \leftarrow 0$;

Se $LP(x) = 0$, então $nx \leftarrow nz + 2$

Senão

$nx \leftarrow nz$

FimSe ;

(* Calcula o Π de cada filho de x *)

Para todo $z \in F_x \setminus \{y\}$, faça

$K_x^{-1} \leftarrow K_x^{-1} - \frac{\lambda(z)}{1-\lambda(z)}$;

Se $nx=0$, então

$P_x \leftarrow P_x \cdot \frac{1-\lambda(z)}{\lambda(\bar{z})}$

Senão

Se $\lambda(\bar{z})=0$ e $nx=1$, então

$P_x \leftarrow \prod_{\substack{t \in F_x \\ t \neq z}} \frac{\lambda(\bar{t})}{1-\lambda(t)} \cdot \frac{LP(x)}{1-LP(\bar{x})}$;

Senão

$P_x \leftarrow 0$;

FimSe

Fimfo ;

(* Cálculo da mensagem que pai x manda para z *)

$$\Pi(z) \leftarrow K.P ;$$

$$\Pi(\bar{z}) \leftarrow 1 - K ;$$

(* Pai x pede que filho z receba sua mensagem Π *)

z MsgdoPai msg: $\Pi(z), \Pi(\bar{z})$; dest.: z; orig.: x ;

FimPara (* todo z *)

(d) MandaMsgPai ;

$$\left[K \right] ;$$

$$K^{-1} \leftarrow 1 - L(x)F(\bar{x}) - L(\bar{x})F(x) ;$$

$$\lambda(x) \leftarrow K \left[L(x) \left[1 - F(\bar{x}) \right] + F(\bar{x}) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

$$\lambda(\bar{x}) \leftarrow K \left[L(\bar{x}) \left[1 - F(x) \right] + F(x) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

(* Pede para pai p receber mensagem do filho x *)

p MsgdeFilho msg: $\lambda(x), \lambda(\bar{x})$; dest.: p; orig.: x ;

(e) CrençaGlobal

$$K^{-1} \leftarrow 1 - \lambda(x)\Pi(\bar{x}) - \lambda(\bar{x})\Pi(x) ;$$

$$C(x) \leftarrow K \left[\lambda(x) \left[1 - \Pi(\bar{x}) \right] + \Pi(\bar{x}) \left[1 - \lambda(x) - \lambda(\bar{x}) \right] \right] ;$$

$$C(\bar{x}) \leftarrow K \left[\lambda(\bar{x}) \left[1 - \Pi(x) \right] + \Pi(x) \left[1 - \lambda(x) - \lambda(\bar{x}) \right] \right] ;$$

(f) CombinaLocal e Π ;

$$\left[K \right] ;$$

(* combina a crença Local com a mensagem do Pai *)

$$K^{-1} \leftarrow 1 - L(x)\Pi(\bar{x}) - L(\bar{x})\Pi(x) ;$$

$$LP(x) \leftarrow K \left[L(x) \left[1 - \Pi(\bar{x}) \right] + \Pi(\bar{x}) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

$$LP(\bar{x}) \leftarrow K \left[L(\bar{x}) \left[1 - \Pi(x) \right] + \Pi(x) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

(g) NovaEvidência msg.: $E(x), E(\bar{x})$; dest.: x; orig.: usuário

$$\left[1, 1', K \right] ; \quad (* \text{ Variáveis temporárias } *)$$

(* Atualiza a crença local *)

$$K^{-1} \leftarrow 1 - L(x)E(\bar{x}) - L(\bar{x})E(x) ;$$

$$1 \leftarrow K \left[L(x) \left[1 - E(\bar{x}) \right] + E(\bar{x}) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

$$1' \leftarrow K \left[L(\bar{x}) \left[1 - E(x) \right] + E(x) \left[1 - L(x) - L(\bar{x}) \right] \right] ;$$

$L(x) \leftarrow 1$;
 $L(\bar{x}) \leftarrow 1'$;
 Minha CombinaLocale Π ;
 Minha MandaMsgFilhos msg.: ϕ ; (* para todos os filhos *)
 Minha MandaMsgPai ;
 Minha CrençaGlobal ;

(h) MsgdoPai msg: $P(x), P(\bar{x})$; dest.: x; orig.: p ;
 $\Pi(x) \leftarrow P(x)$
 $\Pi(\bar{x}) \leftarrow P(\bar{x})$
 Minha CombinaLocale Π ;
 Minha MandaMsgFilhos msg.: ϕ ; (* para todos os filhos *)
 Minha CrençaGlobal (* Atz a crença global *)

(i) QualCrença msg.: - ; dest.: x ; orig.: y .
 Entregar msg: $C(x), C(\bar{x})$; dest.: y ; orig.: x ;

Não foi definido o gabarito da classe usuário e nem especificado o sistema de suporte. Contudo a classe usuário pode ter um gabarito com apenas duas operações: ColetaEvidência e MaiorCrença. A cada evidência nova que surge, ColetaEvidência deve ver qual nó discerne aquela evidência, definir a função de suporte simples correspondente, e chamar NovaEvidência daquele nó (a identificação do nó precedendo o nome da operação, indica que a operação é do nó identificado. Se a palavra "Minha" precede o nome de uma operação, então a operação invocada é a do próprio objeto invocador). Já a operação MaiorCrença pode especificar um subconjunto de nós, e pedir a crença de cada um deles, via a operação QualCrença, e então, através de uma função de perda, tomar alguma decisão.

O sistema de suporte pode variar bastante em função do contexto topológico, recursos computacionais disponíveis e da aplicação. Mas, no Mínimo, ele deve ser capaz de ativar os objetos (nós da rede) especificados no modelo, receber e entregar mensagens aos objetos sem ultrapassagens. Isto é, deve haver uma disciplina de fila, tal, que garanta que uma mensagem postada primeiro para um determinado objeto chegue antes do que outra mensagem para o mesmo objeto, postada mais tarde.

6.4 ANÁLISE DE COMPLEXIDADE

Nesta seção é feita a análise de complexidade de tempo e espaço para cada classe de objetos (nós), em função do fator de ramificação máximo encontrado na árvore de hierarquia, e do total de fss do conjunto ζ propagadas pela árvore que direta ou indiretamente impactou o nó sob análise.

6.4.1 NÓ FOLHA

Um nó folha x recebe uma mensagem para cada fss do conjunto ζ_x ou do conjunto ζ_x^+ . As mensagens oriundas das fss do conjunto ζ_x são tratadas pela operação NovaEvidência de x . Para cada uma destas fss tem-se o seguinte esforço:

- . 8 atribuições ;
- . 20 somas ; e,
- . 12 produtos.

Este esforço pode ser representado por uma constante c_1 . Considerando todas as fss em ζ_x , tem-se que o esforço é $|\zeta_x|c_1$.

As mensagens oriundas das fss do conjunto ζ_x^+ são tratadas pela operação MsgdoPai. E para cada fss de ζ_x^+ , tem-se o seguinte esforço:

- . 5 atribuições ;
- . 10 somas ; e,
- . 9 produtos .

Seja c_2 a constante que representa este esforço, então o esforço para todas as fss em ζ_x^+ é $|\zeta_x^+|c_2$. Somando o esforço dado pelas fss dos dois conjuntos ζ_x e ζ_x^+ , tem-se $|\zeta_x|c_1 + |\zeta_x^+|c_2$. Fazendo $c = \max(c_1, c_2)$, vem que a complexidade de tempo em um nó folha é linear na quantidade de fss disponíveis para o sistema, isto é, $T(\text{folha}) = O(|\zeta|)$. Já que $\zeta = \zeta_x \cup \zeta_x^+$.

O gabarito de dados fixos de um nó folha é composto das crenças L, C e II, o que perfaz 6 posições reais de memória. As áreas adicionais gastas por cada uma de suas Operações em termos de variáveis locais é no máximo 3 posições reais. Portanto a complexidade de tempo para as instâncias de nós folhas é uma constante. Assim $S(\text{folha}) = O(1)$.

6.4.2 Nó RAIZ

O nó raiz recebe uma mensagem para cada fss do conjunto ζ . A operação `MsgdeFilho` do objeto raiz trata cada uma destas mensagens como o seguinte esforço:

- . $2|F_r| + 5$ atribuições ;
- . $4|F_r| + 7$ somas ; e,
- . $2|F_r| + 5$ produtos .

Isto pode ser representado por $c_1|F_r| + c_2$. Em adição, apenas uma mensagem entre todas as de ζ , pode acrescentar mais um esforço de $c_2|F_r|$ produtos, para atualizar o produtório conforme proposição 5 da seção 4. Assim, na pior hipótese, temos $c_1|F_r||\zeta| + c_2|F_r| + c_3$. Tomando f como o maior fator de ramificação na árvore vem que a complexidade de tempo na raiz é $T(\text{raiz}) = O(f|\zeta|)$.

O gabarito de dados fixo para uma instância de nó raiz é composto de uma mensagem λ por nó filho de r . Como cada λ tem duas posições reais, será suposto que cada posição real toma c posições de memória. Tomando f como o maior fator de ramificação da árvore \mathbb{A} , então a área total gasta pelo gabarito fixo é $2cf$.

As áreas adicionais gasta pelas operações, em termos de variáveis locais, não excede a 4 posições reais, o que dá uma área adicional de $4c$. Somando as áreas fixas e adicionais tem-se $c(2f+4)$. Portanto a complexidade de espaço para um nó raiz é $S(\text{raiz}) = O(f)$.

6.4.3 Nó INTERNO

Um nó interno x , $x \neq r$, recebe uma mensagem para cada uma das fss dos conjuntos ζ_x , $\zeta_x^- \setminus \zeta_x$ ou ζ_x^+ .

Para cada mensagem gerada por uma fss do conjunto ζ_x , tratada pela operação `NovaEvidência`, tem-se o seguinte esforço:

- . $5|F_x| + 18$ atribuições ;
- . $5|F_x| + 40$ somas ; e,
- . $5|F_x| + 28$ produtos.

Este esforço pode ser representado por $c_1 f + c_2$, com c_1 e c_2 constantes e f o fator máximo de ramificação em \mathbb{A} . O esforço total para as mensagens geradas pelas fss de ζ_x é $|\zeta_x|(c_1 f + c_2)$.

Para cada mensagem oriunda de uma fss de $\zeta_x^- \setminus \zeta_x^+$, tratada pela operação MsgdeFilho, tem-se o seguinte esforço:

- . $5|F_x| + 18$ atribuições;
- . $7|F_x|$ somas; e,
- . $6|F_x| + 24$ produtos.

Este esforço pode ser representado por $c_3 f + c_4$. E ao considerar todas as fss de $\zeta_x^- \setminus \zeta_x^+$ é obtido o esforço $|\zeta_x^- \setminus \zeta_x^+|(c_3 f + c_4)$.

Para cada mensagem gerada por uma fss de ζ_x^+ , tratada pela operação MsgdoPai, se tem o seguinte esforço:

- . $5|F_x| + 12$ atribuições;
- . $7|F_x| + 20$ somas; e,
- . $6|F_x| + 10$ produtos.

Este esforço pode ser representado por $c_5 f + c_6$. E para todas as fss de ζ_x^+ é obtido o seguinte esforço: $|\zeta_x^+|(c_5 f + c_6)$.

Agora fazendo $c = (c_1, c_2, c_3, c_4, c_5, c_6)$, e tomando o esforço para os três conjuntos se obtém o esforço total a que está submetido um nó interno: $|\zeta_x|(cf + c) + |\zeta_x^- \setminus \zeta_x^+|(cf + c) + |\zeta_x^+|(cf + c) = |\xi|(cf + c)$. Isto dá a complexidade de tempo, $T(\text{Interno}) = O(f|\xi|)$.

O gabarito de dados fixo para a classe de nós interno é constituído das crenças L, F, λ, Π, LP e mais um mensagem λ por filho, cada uma gastando duas posições reais. Tomando c como o número de posições de memória por posição real, vem que a área fixa gasta é $2c(5 + f)$. Onde o fator de ramificação f representa a quantidade de mensagens λ retidas (uma por filho).

As áreas de uso local nas operações não excede a 10 posições reais, o que gera um gasto de $10c$.

Somando as áreas temporárias com as áreas permanentes tem-se a complexidade de espaço para um nó interno, $S(\text{interno}) = O(f)$.

Talvez não seja importante pensar na complexidade de tempo em termos de todas as funções de suporte simples (fss) do conjunto ζ , mas apenas naquelas que simultaneamente são informadas ao sistema, tirando-o do seu estado de equilíbrio. Sejam k o número de tais funções que simultaneamente chegam em k diferentes nós da árvore

de hierarquia. O esforço feito por cada classe de nó na árvore para levar o sistema novamente ao estado de equilíbrio é dado por $O(kf)$, $O(kf)$ e $O(k)$, respectivamente nó raiz, interno e folha.

7 CONCLUSÕES

Os tópicos de interesse, para este artigo, da Teoria Matemática da Evidência (Teoria de Dempster-Shafer, Modelo de Transferência de Crença, ou ainda Função de Crença) foram tratados na seção 2. Lá foram abordados os conceitos de crença, função de crença, massa básica de crença, soma ortogonal, quadros de discernimento, refinamento de quadros de discernimento, quadros de discernimento compatíveis, independência de quadros de discernimento compatíveis, proposições equivalentes entre quadros de discernimento compatíveis, funções de crença consistentes (ou induzidas), discernimento relevante da iteração de evidências, e outros.

Como no projeto dos dois algoritmos foi usado o paradigma de orientação a objetos, foi incluída uma seção com os conceitos gerais desta estratégia de projeto.

Após definir hierarquia de hipóteses, foi enunciado o problema de combinar evidências em um espaço de proposições hierarquizado. Mostrou-se que cada nó na árvore de hierarquia pode ser considerado um objeto e como as mensagens podem ser trocadas nesta comunidade de objetos. Foram formalizadas todas as operações necessárias para a troca de mensagens entre os objetos à luz da Teoria de Dempster-Shafer.

Foram descritos dois algoritmos para combinar evidências em um espaço de hipóteses hierarquizado. Um deles sequencial com complexidade de tempo linear na quantidade de hipóteses. O outro algoritmo é paralelo e tem complexidade linear na quantidade de funções de crenças combinadas e no fator de ramificação da árvore que hierarquiza uma instância do problema.

O algoritmo paralelo guarda uma estreita relação com o trabalho de Pearl(1987), que combina evidências no contexto da teoria Bayesiana, no caso particular em que a função de probabilidade conjunta, das variáveis observáveis e não observáveis do modelo, possa ser fatorada e representada por uma rede simplesmente

cónexa. Como no Trabalho de Pearl, aqui também é possível estender os algoritmos, tanto o sequencial como o paralelo, para resolver o problema de combinar evidências hierarquizadas em redes simplesmente conexas.

REFERENCIAS:

- BARNETT, J.A. Computational methods for a mathematical Theory of evidence. In: Proceeding of IJCAI'81:868-875, 1981.
- BHATNAGAR, R. K. & KANAL, L. N. Handling uncertain information: A review of numeric and Non-numeric Methods. In: KANAL, L.N. & LENNER, J.F. (eds.). *Uncertain in Artificial Intelligewnce*. North-Holland, Elsevier Science, 1986. p. 3-26.
- CONEN, A.T. Data Abstraction, Data Encapsulation and Object Oriented Programming. *ACM SIGPLAN Notices*, 19(1):31-35, Jan. 1984.
- CUNNINGHAM, W. & BECK, K. A Diagram for Object-Oriented Programs. *ACM SIGPLAN Notices*, 21(11):361-7, Nov 1986.
- DEMPSTER, A. P. On direct probabilities. *J. Royal Statis. Soc, Ser. B* 25:102-107, 1963.
- DEMPSTER, A. P. New methods for reasoning toward posterior distributions based on sample data. *Ann. Math. Statis.*, 37:355-74, 1967a.
- DEMPSTER, A. P. Upper and lower probabilities induced by a multivalued mapping. *Ann. Math. Statis.* 38:325-39, 1967b.
- DEMPSTER, A. P. Upper and lower probability inferences based on a sample from a finite univariant population. *Biometrika*, 54:515-28, 1967c.
- DEMPSTER, A. P. Upper and lower probabilities generated by a random closed interval. *Ann. Math. Statis.* 39(3):957-56, 1968a.
- DEMPSTER, A. P. A generalization of bayesian inference. *J. Roy. Statis. Soc. Ser B* 30:205-47, 1968b.
- DEMPSTER, A. P. Upper and lower probability inferences for families of hypoteses with monotone density ratios. *Ann Math. Statis.* 40:953-69, 1969.
- DUBOIS, D. & PRADE, H. Combination and propagation of uncertainty with belief functions. In: *Proceedings of IJCAI'85*, 1985. p. 111-13.

- GARVEY, T.D. et alii. An inference technique for integrating knowledge from disparate sources. In: Proceeding of IJCAI'81. 1981. p.319-25.
- GORDON, J. & SHORTLIFE, E.H. The Dempster-Shafer theory of evidence. In: BUCHANAN, B.G & SHORTLIFE, E.H. (eds.). Rule-based expert systems. California, Addison-Wesley, 1984. p. 272-94.
- GORDON, J. & SHORTLIFE, E.H. A method for managing evidential reasoning in hierarchical hypothesis space. Artificial Intelligence, 26(3):323-57, Jul 1985.
- HEWITT, C. et alii. A Universal Modular Actor Formalism for Artificial Intelligence. In: Proceedings of 3rd IJCAI, 1973.
- JONATHAN, M. Orientação para Objetos em SMALTALK-80: Uma abordagem eficaz para a construção de sistema de software. In: Primeiro Simpósio Brasileiro de Engenharia de Software, Itaipava, 1987. p.32-44.
- LEE, Chia-Hoang. A Comparison of two evidential reasoning schemes. Artificial Intelligence, 35(1):127-34, May 1988.
- LEMMER, J.F. Confidence factors, empirism and Dempster-Shafer theory of evidence. In: KANAL, L.N. & LEMMER, J.F. (eds.). Uncertain in Artificial Intelligence. North-Holland, Elsevier Science, 1986. p. 117-26.
- MATICH, G. H. Princípios e Polêmicas sobre a "Orientação a Objetos". Trabalho de pesquisa, Informática/PUC-RJ, 1988.
- NGUYEN, Van et alii. A Generalized Object Model. ACM SIGPLAN Notices, 21(10):78-87, Oct 1986.
- ROSSI, G. Programacion Orientada a Objetos: Impacto en Tecnicas de Descripcion de Software baseados en Rede de Petri. Projeto ETHOS, II EBAT, Petrópolis, Abril 1987.
- ROTENBERG, H.B. Programação Orientada a Objetos-Um enfoque de Engenharia de Software. Tese de Mestrado, Informática/PUC-RJ, Abril de 1987.
- SENNE, E.L.F. Uma abordagem para o tratamento de incerteza em sistemas especialistas. In: Anais do CNMAC'87. Set 1987. p. 846-51.
- SHAFER, G. A mathematical theory of evidence. Princeton University Press, 1976.
- SHAFER, G. Probability judgment in artificial intelligence. In: KANAL, L.N. & LEMMER, J.F. (eds.). Uncertain in Artificial Intelligence. North-Holland, Elsevier Science, 1986. p. 127-36.

- SHAFER, G. & LOGAN, R. Implementing Dempster's Rule for hierarchical evidence. *Artificial Intelligence*, 33:271-98, 1987.
- SILVA, W.T.da & RICHTER, G. Formalização da abordagem orientada a objetos em redes de Petri. In: VELOSO, Paulo A.S.(ed). *Monografias em Ciências da Computação*, No. 4/89, INF-PUC/RJ, Rio de Janeiro, Abril 1989.
- SILVA, W.T.da & SCHWABE, D. Metodos de propagar incerteza em sistemas especialistas. *Relatório técnico*, INF/PUC/RJ, Rio de Janeiro, 1988.
- SILVA, W.T.da & MILIDIÚ, R.L. Algoritmos para Combinação de Evidências. *Relatório Técnico*, DI-PUC/RJ, Rio de Janeiro, 1989a. (A sair em VELOSO, Paulo A.S.(ed). *Monografias em Ciências da Computação*, INF-PUC/RJ, Rio de Janeiro)
- SILVA, W.T.da & MILIDIÚ, R.L. Combinando funções de crença com uma função bayesiana. In: *Anais do XXII Simpósio Brasileiro de Pesquisa Operacional (XXII SBPO)*, Fortaleza, Outubro, 1989b.
- SILVA, W.T.da & MILIDIÚ, R.L. Combinando crenças politômicas em um espaço de proposições comum. In: *Anais do VI Simpósio Brasileiro em Inteligência Artificial (VI SBIA)*, Rio de Janeiro, Novembro, 1989c.
- SMETS, Philippe. Belief functions versus probability functions. *Lecture Notes in Computer Science*, vol 313:17-24, 1988.
- STROM, R. A comparison of the Object-Oriented and Process Paradigms. *ACM SIGPLAN Notices*, 21(10):88-97, Oct 1986.
- TAKAHASHI, T. Introdução a programação Orientada a Objetos. III EBAI, Curitiba, Janeiro de 1988. 148 pp.
- THOMPSON, T.R. Parallel formulation of evidential-reasoning theories. In: *Proceedings of IJCAI'85*, 1985. p. 321-27.
- TOKORO, M. & ISHIKAWA, Y. Concurrent Programming in Orient84/K: An Object-Oriented Knowledge Representation Language. *ACM SIGPLAN Notices*, 21(10):39-48, Oct. 1986.
- ZADEH, L.A. A simple view of the Dempster-Shafer theory of evidence and its implication. *The AI Magazine*, Summer:85-90, 1986.