

**ÁRVORES DE EVENTOS
EM ANÁLISE DE CONFIABILIDADE**

Liliam Ayako Matsunaga

Dissertação apresentada
ao
Instituto de Matemática e Estatística
da
Universidade de São Paulo
para obtenção do grau de Mestre
em
Estatística

Área de Concentração: **TEORIA DA CONFIABILIDADE**
Orientador: **Prof. Dr. WAGNER DE SOUZA BORGES**

–São Paulo, Janeiro de 1991–

Aos meus pais e aos meus sobrinhos
Alexandre, Andréa, Juliana e Renato

Agradecimentos

Agradeço a todos que de forma direta ou indireta contribuíram para a realização deste trabalho¹. Em especial, desejo agradecer:

- Ao Prof. Dr. Wagner de Souza Borges, por tudo que me ensinou, pela paciência e dedicação com que me orientou, pelo apoio em todos os momentos difíceis e, principalmente, pela contribuição substancial que deu a este trabalho.
- À Antônio Souza Vieira Neto (Chefe da Seção de Análise de Risco - Divisão de Engenharia de Segurança Nuclear - COPESP I), pelo apoio, pelo incentivo diário e pela amizade, que foram fundamentais na finalização deste trabalho.
- À Enaldo Pires Montanha, pela dedicação e paciência com que fez todos os desenhos² deste trabalho.
- À Geraldo Toledo Arruda Júnior (Chefe da Divisão de Eletrônica - COPESP II) e Waldomiro Luís Rios de Mello (Chefe da Seção de Produto e Confiabilidade - Divisão de Eletrônica - COPESP II), pela paciência e compreensão demonstradas.
- Ao Prof. Dr. Wilton de Oliveira Bussab, pela contribuição importante que deu a minha formação, me transmitindo, durante os dois anos que trabalhamos juntos, sua tão ampla visão e experiência na estatística aplicada .
- Ao Prof. Dr. Dalton Francisco de Andrade, por ter sido o responsável pela minha vinda para o IME-USP, pelo apoio, pelo incentivo e pelas críticas.
- Aos colegas da COPESP, pelo apoio e amizade. Em especial, à Patrícia da Silva Pagetti (Seção de Análise de Risco).
- À CAPES e a EMBRAPA, pela bolsa que me concederam para fazer o mestrado.

¹A edição deste trabalho foi realizada no Sistema \LaTeX .

²Todos os desenhos contidos neste trabalho foram realizados no Sistema de CAD-CAM "MicroStation PC" da INTERGRAF.

Índice

Capítulo 0	Introdução e Resumo	1
Capítulo 1	Construção de Árvores de Eventos	3
1.1	Introdução	3
1.2	Princípios Básicos	3
1.3	Portas Lógicas e Eventos Especiais	20
Capítulo 2	Análise Qualitativa	25
2.1	Introdução	25
2.2	Elementos Algébricos	26
2.3	Base Completa	30
2.3.1	Algoritmos para a Obtenção de Disjunções e Conjunções Normais	31
2.3.2	Algoritmos para a Obtenção da Base Completa	43
2.3.3	Cortes Espúrios	72
2.4	Bases Irredundantes	73
Capítulo 3	Análise Quantitativa	83
3.1	Introdução	83
3.2	Comportamento Dinâmico de Eventos	85
3.2.1	Eventos Terminais Básicos	86
3.2.2	Eventos Suficientes Mínimos	96
3.2.3	Evento Topo	101
	Referências Bibliográficas	121

Capítulo 0

Introdução e Resumo

A análise de árvores de eventos é um dos principais métodos de análise de confiabilidade de sistemas. Este método foi inspirado nas representações gráficas de circuitos eletrônicos lógicos e é utilizado, em geral, para avaliar, sob diversos pontos de vista, estados do sistema através da modelagem destes últimos a partir de combinações de estados de diversos componentes desse sistema. Embora seja um método de natureza geral, sua aplicação mais comum se dá no contexto da análise de sistemas complexos de engenharia, onde é utilizado como ferramenta para: (a) a análise de desempenho de sistemas; (b) a avaliação de alterações em projetos; (c) a verificação de conformidade com requisitos de confiabilidade ou segurança; e (d) o estudo de políticas de manutenção.

O método de análise de árvores de eventos foi introduzido em 1961 por H. A. Watson [Bel] da Bell Telephone Laboratories, durante o estudo da segurança do Sistema de Controle de lançamento dos mísseis Minuteman da força aérea norte-americana. Os primeiros artigos publicados sobre este método foram apresentados no simpósio de segurança realizado em 1965, sob o patrocínio da Universidade de Washington e da Boeing Aerospace Company, que marcou o início do interesse generalizado por sua utilização em diversos ramos da indústria (cf. [CB], [Lam]). No início da década de 70, softwares de aplicação começaram a ser desenvolvidos para possibilitar a realização da análise de árvore de eventos através de computadores (cf. [VN], [FHM], [Fus]) e difundir sua utilização.

Uma análise de árvore de eventos consiste fundamentalmente das etapas de construção, análise qualitativa e análise quantitativa, que podem ser assim resumidas: a construção da árvore de eventos é o processo lógico dedutivo que, a partir de um evento pré-definido, denominado evento topo, procura identificar as várias combinações lógicas de eventos que contribuem para a sua ocorrência; a análise qualitativa da árvore de eventos consiste basicamente da avaliação da arquitetura do sistema através da determinação de todas as causas fundamentais de ocorrência do evento topo; e a análise quantitativa, procura estabelecer medidas de desempenho que traduzam características do comportamento dinâmico de alguns eventos especiais da árvore.

Devido as suas raízes históricas, nas aplicações mais comuns desse método, as relações causais entre os eventos de uma árvore são modeladas somente em termos de eventos que descrevem falhas de componentes integrantes do sistema sob avaliação. Neste contexto, árvores de eventos são conhecidas como árvores de falhas. Porém, existem certas situações em que para modelar o comportamento lógico de um sistema de forma mais detalhada, é necessário considerar relações causais não só entre eventos que descrevem falhas, mas também entre eventos que descrevem o comportamento correto de componentes e/ou situações de conformidade com procedimentos operacionais do sistema. Estes tipos de relações são consideradas, para se evitar a necessidade da realização de uma pré-análise subjetiva da árvore, a fim de verificar a possível existência de condições incompatíveis que podem levar a obtenção de causas fundamentais de ocorrência espúrias para o evento topo, isto é, causas fundamentais de ocorrência que não tenham significado físico ou que são impossíveis de ocorrer por envolverem eventos mutuamente exclusivos, ou por representarem situações de não conformidade com o procedimento operacional do sistema.

Neste trabalho, procurar-se-á montar a conceituação matemática teórica do modelo de árvore de eventos utilizando como base a teoria dos conjuntos, e apresentar de forma unificada a teoria básica encontrada na literatura, corrigindo aspectos notacionais e evitando o excesso de informalidade com que são apresentados alguns resultados teóricos.

Capítulo 1

Construção de Árvores de Eventos

1.1 Introdução

Neste capítulo é apresentada a técnica de construção dos modelos que constituem a base fundamental do método de árvore de eventos para a análise de confiabilidade.

1.2 Princípios Básicos

Uma árvore de eventos é construída com base na documentação de projeto do sistema de engenharia de interesse, em consultas aos projetistas e ao pessoal responsável pela operação do sistema e em levantamentos destinados a avaliar os efeitos de cada estado possível de cada componente no desempenho do sistema (Análise de Modos de Falha e Efeitos - AMFE [HK]).

Para motivar a apresentação dos elementos básicos para a construção de árvores de eventos, considere o sistema de alimentação de emergência descrito no Exemplo 1.2.1

abaixo.

Exemplo 1.2.1

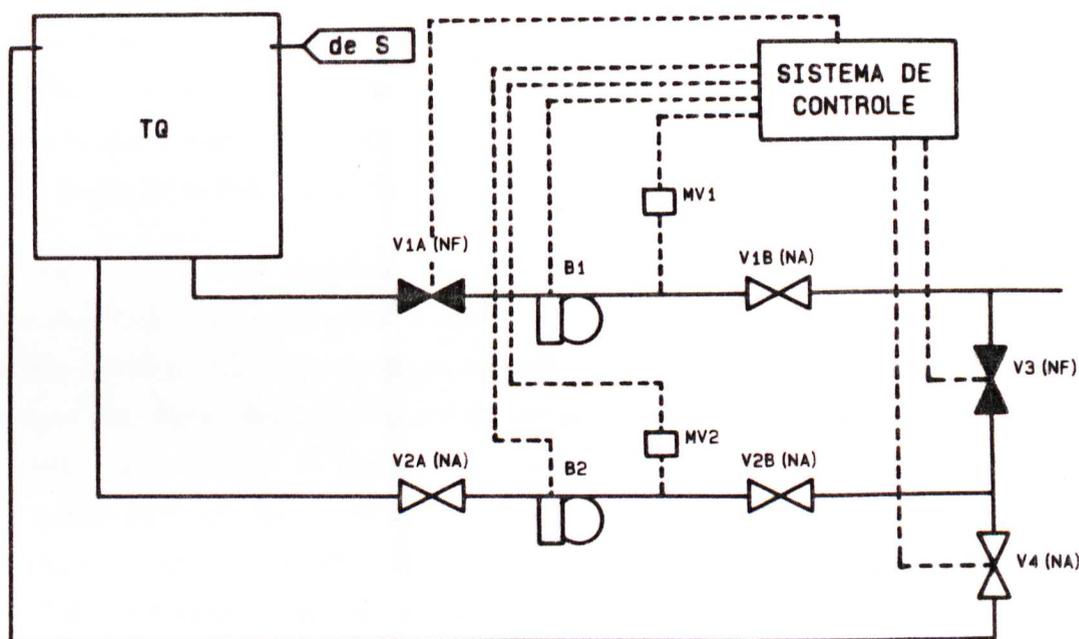


Figura 1.2.1: Sistema de alimentação de emergência

Descrição funcional:

O sistema mostrado na Figura 1.2.1 opera em regime de “stand-by”, e quando é requisitado, um sinal do sistema de controle abre a válvula V1A (normalmente fechada - NF), dá partida nas moto-bombas B1 e B2 e ajusta suas vazões com a informação recebida pelos medidores de vazão MV1 e MV2. O controle da vazão das bombas é feito pelo sistema de controle, através da liberação/supressão de corrente para o motor.

Caso não ocorra a alimentação, um sinal do sistema de controle desliga B1, fecha V1A e inverte o estado das válvulas V3 (normalmente fechada - NF) e V4 (normalmente aberta - NA), numa tentativa de efetuar a alimentação através da bomba redundante B2.

As informações adicionais fornecidas são que: (a) a água de alimentação encontra-se armazenada no tanque TQ e é fornecida por um outro sistema S independente;

(b) as válvulas manuais V1B, V2A e V2B (normalmente abertas - NA) são utilizadas para isolamento das bombas em caso de manutenção; (c) as moto-bombas B1 e B2 sofrem manutenções periódicas, com o procedimento prescrevendo que somente uma bomba por vez deve ser isolada para manutenção, estando a outra disponível, caso o sistema seja requisitado.

Uma questão de interesse na avaliação do desempenho deste sistema é sem dúvida a probabilidade do sistema de alimentação de emergência não entrar quando requerido. Obviamente existem outras questões de interesse nessa avaliação, mas considere a primeira para que se possa introduzir as idéias básicas.

Para responder a esta questão, um analista de confiabilidade recomendará, normalmente, que se construa uma árvore de eventos, isto é, uma descrição formal das relações de causa e efeito entre eventos que produzem como resultado final a não entrada do sistema quando requerido. Este resultado final é denominado evento topo da árvore, que é construída, a partir dele, através de um processo lógico "backward" de refinamento dedutivo. Assim, na primeira etapa do processo, procura-se determinar os eventos, e a combinação lógica destes, cuja ocorrência equivale à ocorrência do evento topo. Nas etapas seguintes, procura-se fazer o mesmo com cada evento novo que é introduzido na árvore nas etapas anteriores. Esse processo de refinamento de um evento termina quando ele se refere a um estado de um componente, ou quando não é possível ou não se tem interesse nesse refinamento. Para representar um evento e seu refinamento em uma árvore de eventos, utiliza-se uma notação gráfica da forma

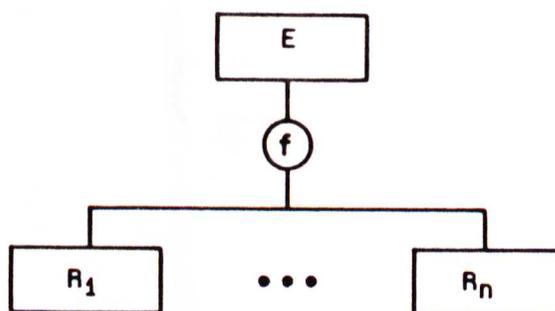


Figura 1.2.2: Refinamento de um evento

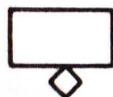
onde E , denominado evento de saída da porta lógica \textcircled{f} , representa o evento que foi refinado e \textcircled{f} , juntamente com os eventos R_1, \dots, R_n , denominados eventos de entrada em \textcircled{f} , representam o refinamento de E . \textcircled{f} , por sua vez, representa uma fórmula com n -variáveis, que aplicada sobre os eventos de entrada, R_1, \dots, R_n , em \textcircled{f} , produz como resultado o evento E .

Para simplificar a terminologia, todo evento de saída E de uma porta lógica \textcircled{f} será denominado evento porta \textcircled{f} , os eventos de entrada em \textcircled{f} serão referidos como entradas de E e a expressão resultante da aplicação de \textcircled{f} à R_1, \dots, R_n será denominada expressão local de E .

Os eventos que representam a finalização de um processo de refinamento são denominados eventos terminais, como sugere o próprio nome, e são representados por símbolos especiais. Assim, enquanto os eventos porta são sempre representados através de retângulos, um evento terminal é representado pelo símbolo



e é denominado evento terminal básico, quando se referir a um evento cuja frequência de ocorrência será quantificada estatisticamente. Caso contrário, o símbolo



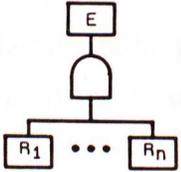
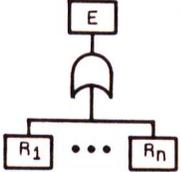
é utilizado.

Eventos terminais representados desta última forma são denominados eventos terminais não refinados e se referem, em geral, a eventos cujas possíveis causas são estatisticamente desprezíveis, ou nos quais não se tem interesse, ou para os quais não se dispõe das

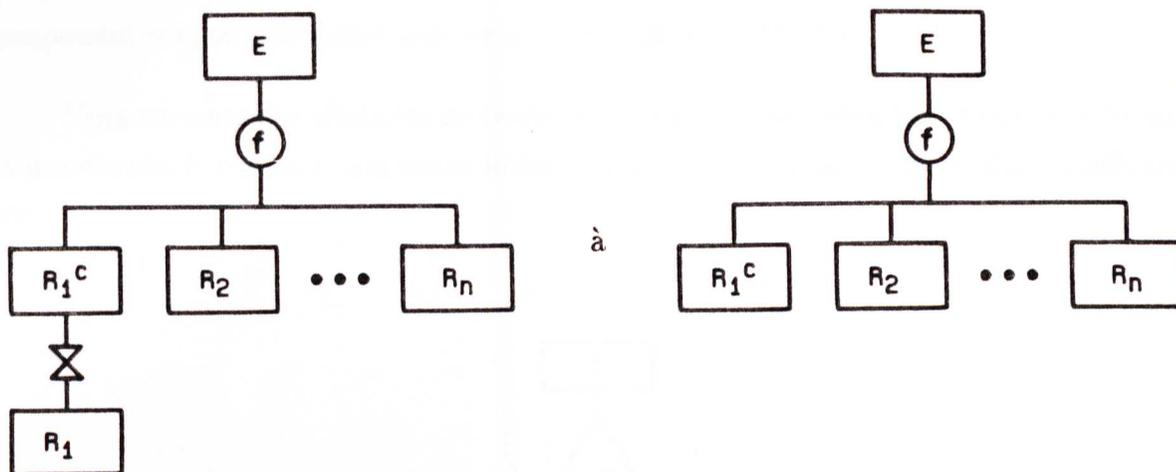
informações que possibilitem seu refinamento ou a quantificação estatística de sua frequência de ocorrência.

Uma porta lógica \mathcal{L} pode representar qualquer expressão lógica. Apesar deste fato, costuma-se utilizar, na prática, portas lógicas de natureza mais simples. Estas portas lógicas são denominadas fundamentais, e são dadas na tabela 1.2.1, juntamente com a simbologia utilizada nesses casos.

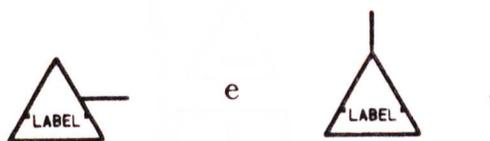
Tabela 1.2.1: Símbolos de portas lógicas fundamentais

Nome da porta \mathcal{L}	Notação gráfica	Expressão local	Relação causal
“AND”		$E = R_1 \cap \dots \cap R_n$	o evento de saída E ocorre se todos os eventos de entrada R_1, \dots, R_n ocorrerem simultaneamente
“OR”		$E = R_1 \cup \dots \cup R_n$	o evento de saída E ocorre se pelo menos um dos eventos de entrada R_1, \dots, R_n ocorrer
“NOT”		$E = R_1^c$	o evento de saída E ocorre se o evento de entrada R_1 não ocorrer

Quando a construção de uma árvore envolve a utilização de eventos que representam condições operacionais corretas e/ou situações indicando a disponibilidade do componente para o sistema, a convenção, em geral, adotada na literatura, e que também será adotada neste trabalho, é representar estes eventos na árvore, como saída de uma porta lógica “NOT” (Tabela 1.2.1). Assim, prefira simbolicamente

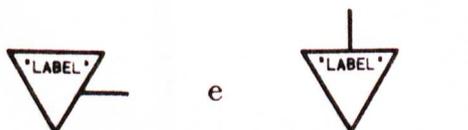


Além da simbologia utilizada para eventos e portas lógicas, outros símbolos são também utilizados para facilitar a construção de uma árvore de eventos. Dois símbolos úteis e muito utilizados são os símbolos de transferência idêntica



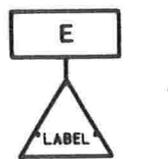
que são utilizados quando um trecho da árvore aparece repetido a partir de um certo ponto. O símbolo \triangleleft é utilizado para identificar o ponto de partida do trecho repetido; e o símbolo \triangleup , para identificar onde é repetido o trecho iniciado em \triangleleft .

Quando se trata da repetição de trechos análogos, isto é, com a mesma lógica mas envolvendo componentes e/ou condições distintas, costuma-se utilizar os símbolos de transferência análoga

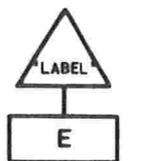


Nestes casos, os componentes e/ou condições que serão alterados no trecho iniciado em ∇ para definir o trecho repetido em ∇ , devem aparecer, entre parênteses, ao lado dos componentes e/ou condições que serão alterados deste trecho.

Uma variante dos símbolos de transferência é utilizada para indicar que o refinamento de um evento E continua em outra folha ou outro lugar. Isto é representado graficamente por



e a continuação do refinamento de E se dá então em outro ponto, a partir do símbolo



Para exemplificar a construção de uma árvore de eventos, considere o problema do sistema de alimentação de emergência, descrito no Exemplo 1.2.1.

Exemplo 1.2.1 (continuação) A Figura 1.2.3 contém a árvore de eventos para o evento topo “sistema de alimentação de emergência não entra quando requerido”.

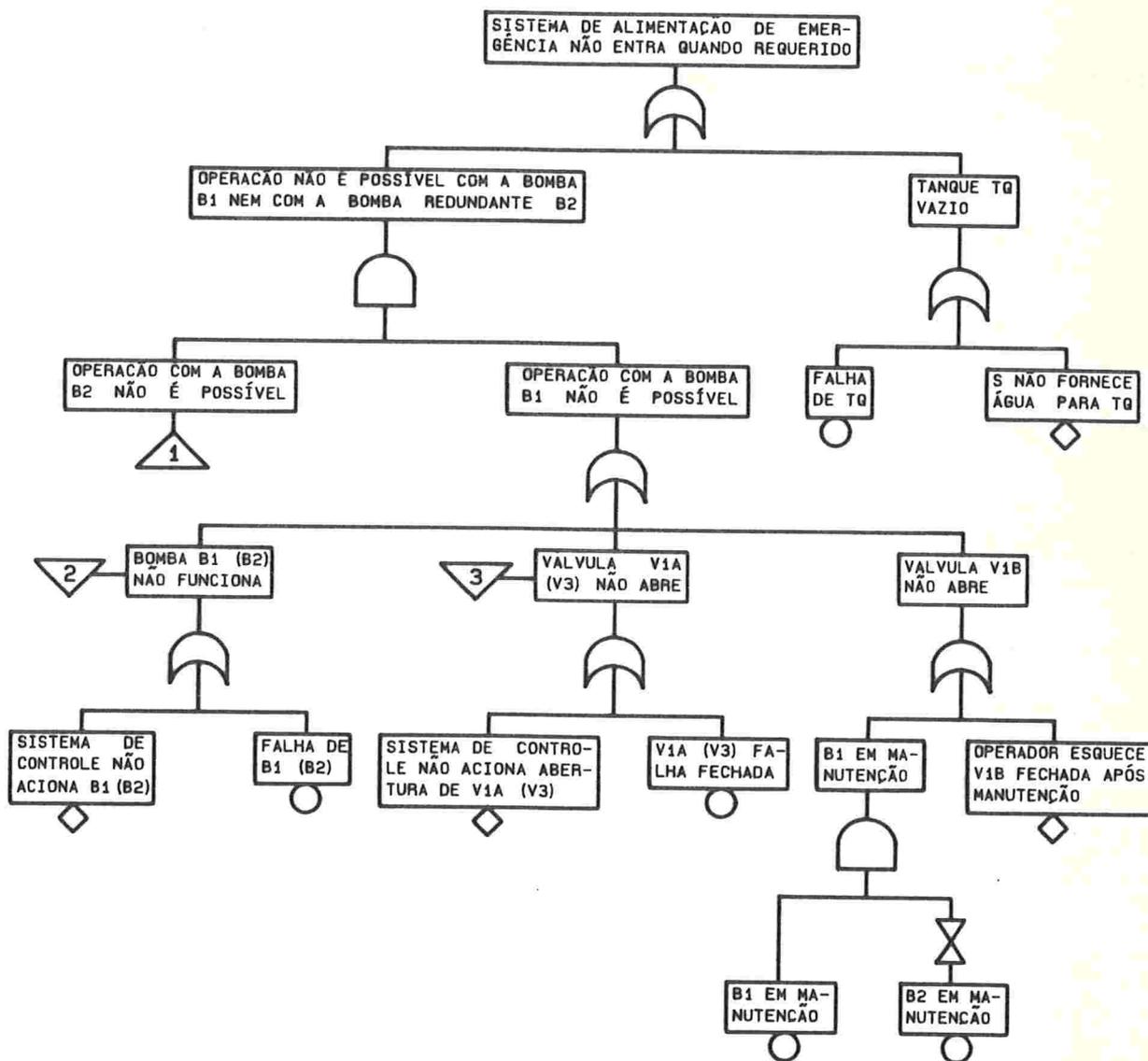


Figura 1.2.3: Árvore de eventos para o evento topo “sistema de alimentação de emergência não entra quando requerido”

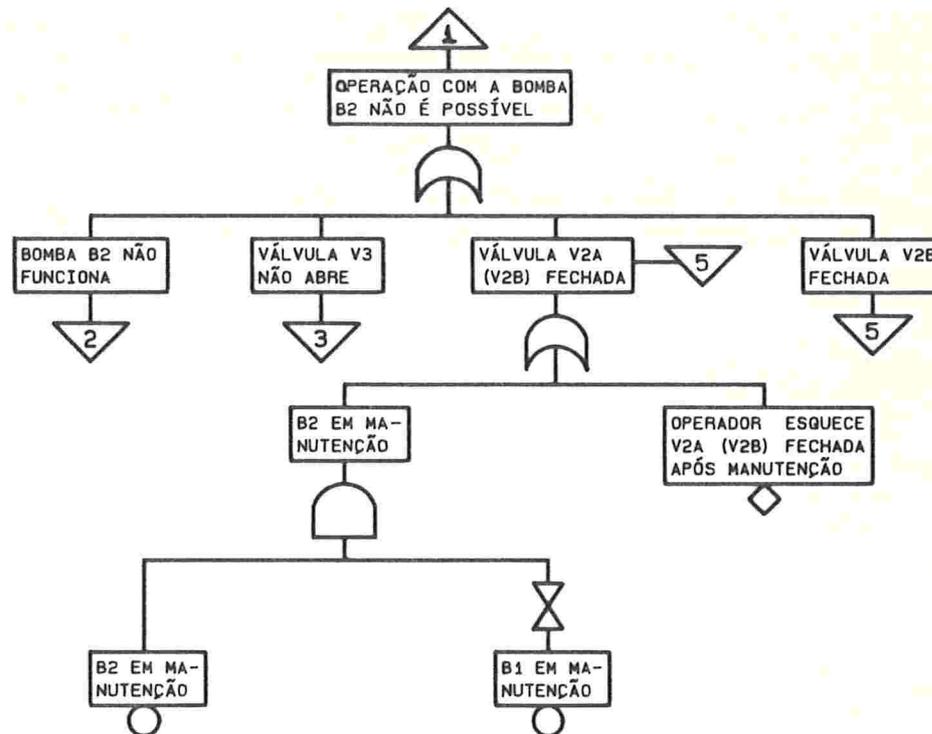


Figura 1.2.3: (continuação)

Apesar da simplicidade do Exemplo 1.2.1, não é difícil perceber que a construção de uma árvore de eventos é, em geral, uma etapa demorada e crítica na análise de confiabilidade, uma vez que é necessário conhecer profundamente o projeto do sistema de interesse, a função que o sistema deve desempenhar e as inter-relações entre os seus componentes. Além disso, como se trata de um modelo estático, sua utilização no estudo de processos dinâmicos requer ainda a identificação de condições de contorno para a análise tais como: a) condição de operação do sistema (partida; em regime; parada; transitórios, etc ...); e b) estado inicial de equipamentos e componentes para a condição de operação do sistema considerada.

Outros fenômenos que podem ser estudados através de modelos de árvores de eventos estão descritos nos Exemplos 1.2.2 e 1.2.3 a seguir, que serão utilizados neste trabalho para exercitar a metodologia de análise apresentada.

Exemplo 1.2.2

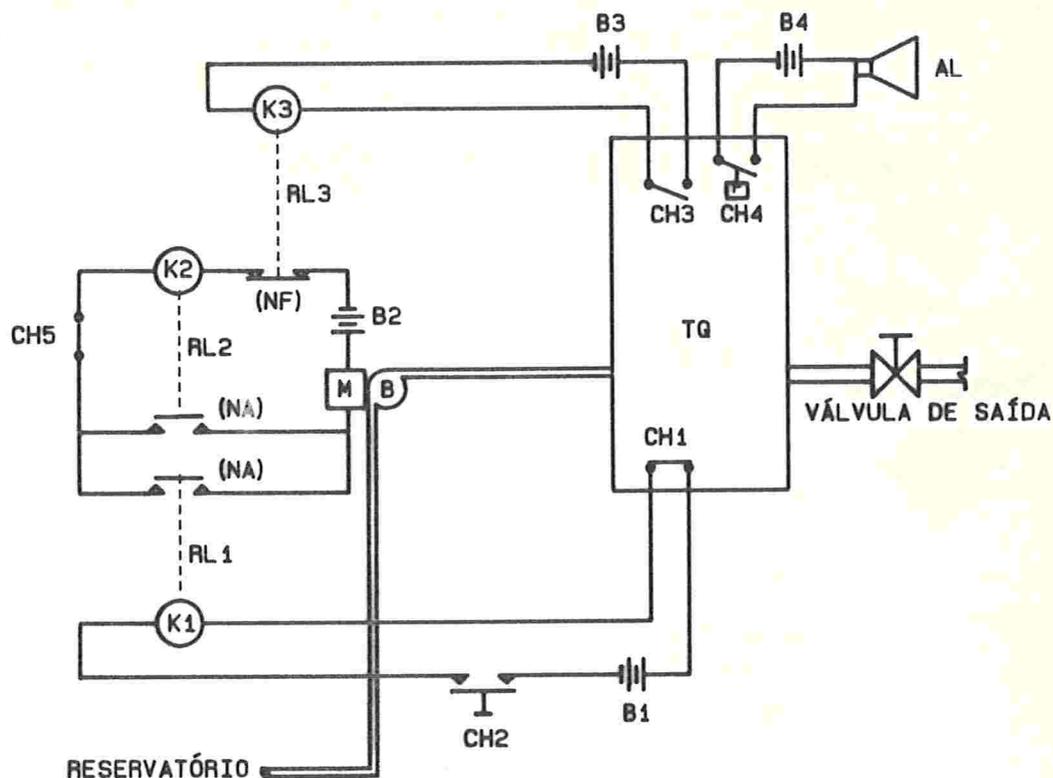


Figura 1.2.4: Sistema de bombeamento

Descrição funcional:

Para encher o tanque de abastecimento TQ, o operador aciona a chave CH2 tipo "push-button". Como a chave de bóia CH1 está inicialmente fechada, a bobina K1 do relé RL1 é energizada causando: o fechamento do contato de RL1 (normalmente aberto - NA); a energização da bobina K2 do relé RL2; o fechamento do contato de RL2 (normalmente aberto -NA); e a partida da moto-bomba MB.

À medida que o tanque enche, a chave de bóia CH1 se abre, desenergizando a bobina K1 e abrindo o contato de RL1. O enchimento do tanque prossegue até o fechamento da chave de bóia CH3, que causa a energização da bobina K3 do relé RL3 e a abertura do contato de RL3 (normalmente fechado - NF). Com isto, a alimentação do motor M é interrompida, cessando o enchimento do tanque TQ. A partir daí, o consumo do líquido é iniciado e, automaticamente, toda vez que o tanque atinge o nível da bóia de CH1, a

moto-bomba reinicia o enchimento do tanque.

Caso este sistema de desligamento automático do motor M falhe, existe uma chave de pressão CH4 que deve fechar, quando a pressão no tanque atingir a pressão de “setpoint” P. O fechamento do contato de CH4 faz soar o alarme AL, que alerta o operador para abrir manualmente a chave CH5, de forma a evitar que a pressão no tanque atinja seu nível de ruptura $P^* > P$.

Um evento topo de interesse para este sistema é “ruptura do tanque TQ”, e sua árvore de eventos é mostrada na Figura 1.2.5 abaixo.

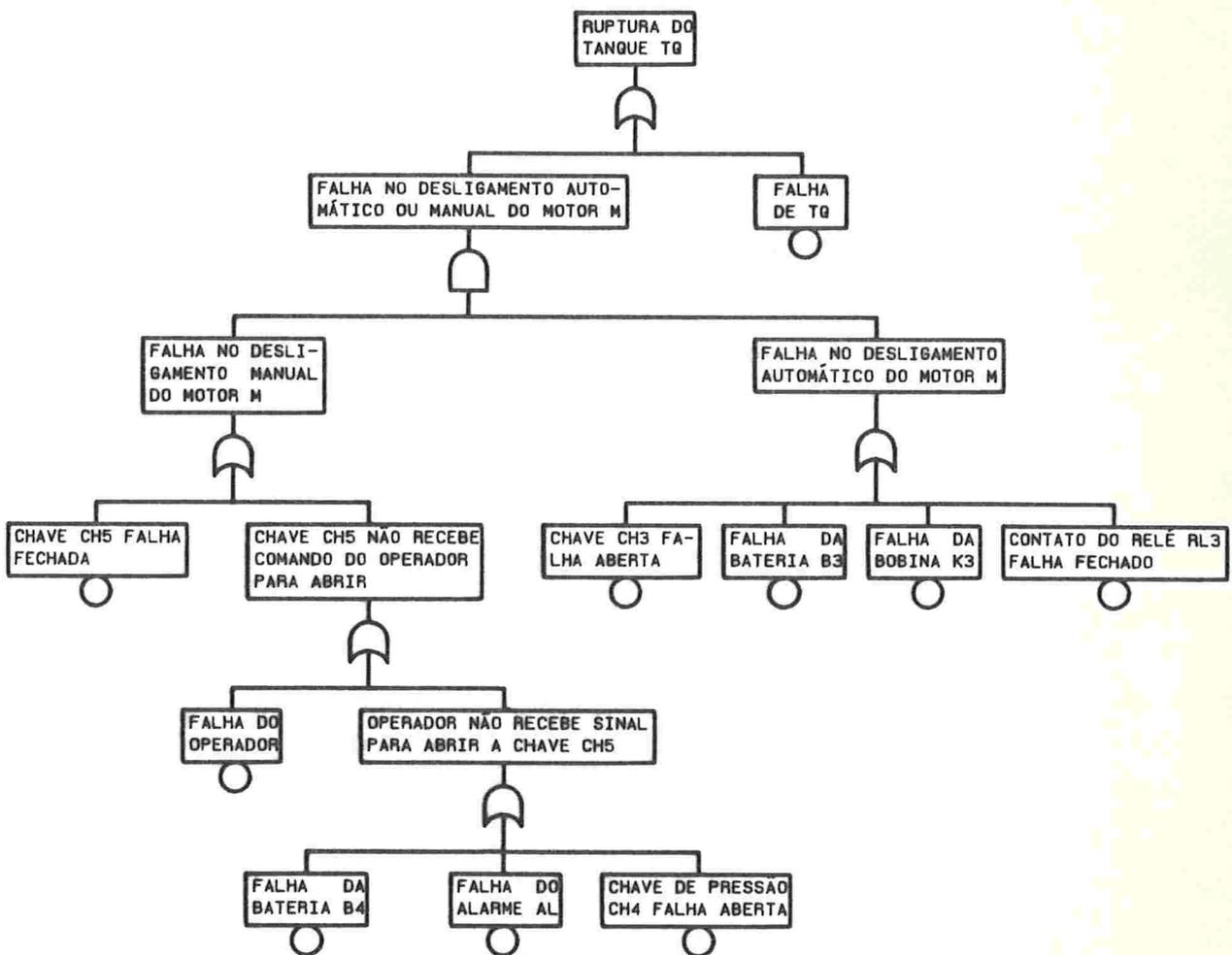


Figura 1.2.5: Árvore de eventos para o evento topo “ruptura do tanque TQ”

Em muitos fenômenos encontrados na prática, as relações causais entre eventos podem ser perfeitamente modeladas utilizando-se somente as portas lógicas “AND” e “OR”. Algumas vezes, este tipo de modelagem resulta de uma simplificação do tipo $A \cup (A^c \cap B) = A \cup B$, que é implicitamente realizada na construção da árvore. Como exemplo de tal situação, considere a árvore de eventos para o evento topo “ruptura do tanque TQ”, referente ao sistema de bombeamento descrito no Exemplo 1.2.2, que é dada na Figura 1.2.5. Nessa árvore, o evento “chave CH5 não recebe comando do operador para abrir” poderia ser refinado da seguinte maneira:

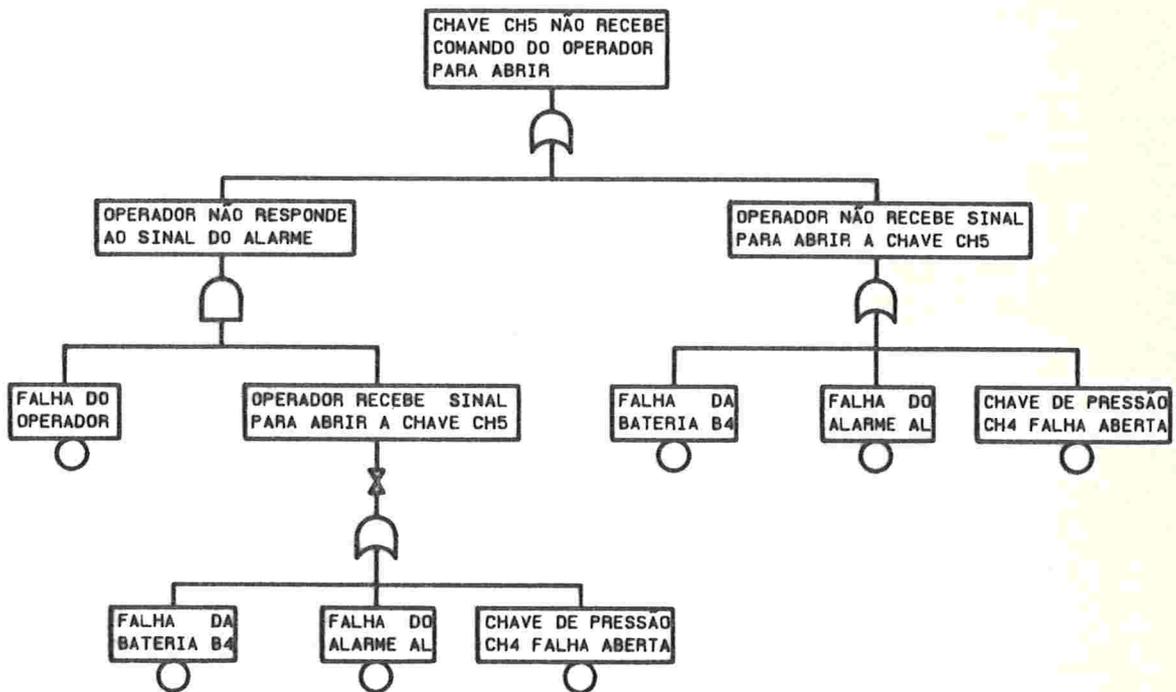


Figura 1.2.6: Refinamento do evento “chave CH5 não recebe comando do operador para abrir”

Entretanto, na construção da árvore utilizou-se a forma equivalente

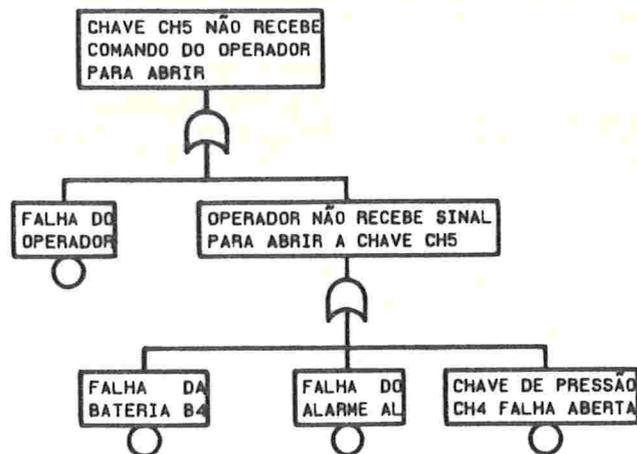


Figura 1.2.7: Refinamento equivalente do evento “chave CH5 não recebe comando do operador para abrir”

Neste caso, a inclusão da porta lógica “NOT” para descrever as relações de dependência entre eventos pode ser simplesmente suprimida, sem perda de informação relevante. Entretanto, existem certas situações em que somente a utilização das portas lógicas “AND” e “OR” não é suficiente para descrever todas as relações de dependência entre os eventos de uma árvore. Tais situações podem ocorrer, por exemplo, quando:

- (a) a coexistência de certos eventos é impossibilitada por restrições de projeto;
- (b) o sistema muda seu modo de operação após um período de tempo determinístico (“phased mission”) e o interesse está na ocorrência de uma falha na segunda fase de operação do sistema. Neste caso, a árvore de eventos para a falha de interesse é do tipo:

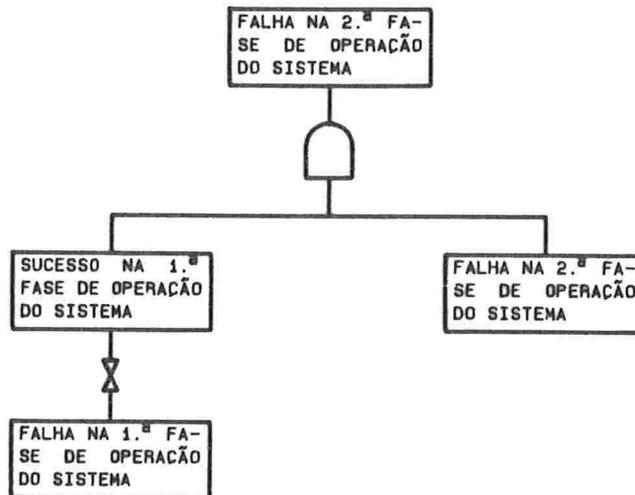


Figura 1.2.8: Árvore de eventos para o evento topo “falha na 2.ª fase de operação do sistema” ;

- (c) o sistema muda seu modo de operação devido a ocorrência de eventos aleatórios (“stochastic phase changes”) e o interesse é modelar características dinâmicas do sistema (problema particularmente aplicável a situações envolvendo sistemas “stand-by”).

Na construção da árvore de eventos para o evento topo “sistema de alimentação de emergência não entra quando requerido”, descrito no Exemplo 1.2.1, encontra-se a situação (a) acima, devido a impossibilidade das moto-bombas B1 e B2 sofrerem manutenção simultaneamente. Como outro exemplo de situação envolvendo a utilização das portas lógicas “AND”, “OR” e “NOT”, considere o sistema de iluminação descrito no Exemplo 1.2.3 a seguir.

Exemplo 1.2.3 (Johnston & Matthews [JM])

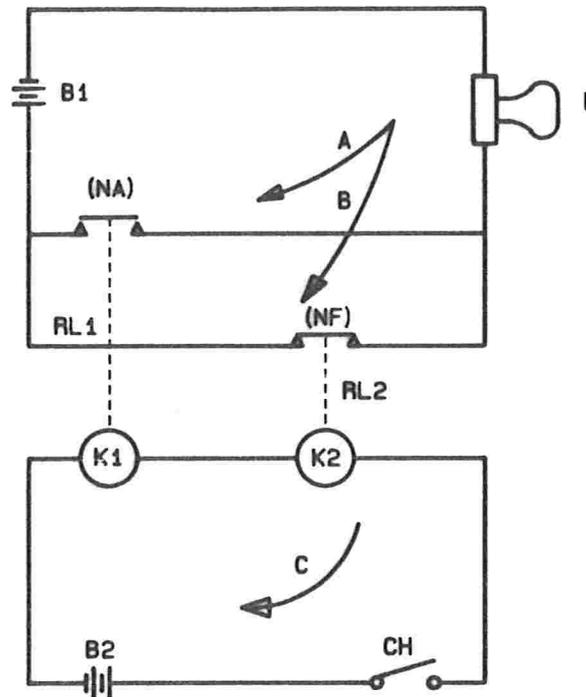


Figura 1.2.9: Sistema de iluminação

Descrição funcional :

O sistema de iluminação dado na Figura 1.2.9 deve operar normalmente com o circuito A. Para tal, o operador deve fechar a chave CH, o que causa a energização das bobinas K1, do relé RL1, e K2, do relé RL2, de forma que o contato de RL1 (normalmente aberto - NA) seja fechado e o contato de RL2 (normalmente fechado -NF) seja aberto.

No caso de uma interrupção do circuito A, para reestabelecer a luz, um subsistema “stand-by”, que opera através do circuito B, pode ser ativado pelo operador, abrindo a chave CH. A abertura de CH causa o fechamento do contato de RL2, devido a desenergização da bobina K2. Deve-se observar que o subsistema “stand-by” também é ativado se a corrente no circuito C for de alguma outra forma interrompida.

Este exemplo se encaixa na situação (c) descrita na página 16. Em situações como esta, é necessário identificar e definir as fases de operação do sistema, os eventos que geram a transição de fase e as possíveis maneiras do sistema falhar em cada fase.

Neste exemplo, existem duas fases de operação:

Fase I - o sistema opera através do circuito A.

Fase II - o sistema opera através do circuito B.

A transição da Fase I para a fase II ocorre de uma das seguintes formas:

- (1) falha no circuito A que leva a ativação manual do circuito B. Situação que ocorre através dos eventos : “falha da lâmpada L”, “falha da bateria B1” e “contato do relé RL1 abre inadvertidamente”;
- (2) falha no circuito C que leva a ativação automática do circuito B. Situação que ocorre através dos eventos : “falha da bobina K1”, “falha da bobina K2”, “falha da bateria B2” e “chave CH abre inadvertidamente”.

Admitindo que a falha do operador é desprezível e que o sistema está operando através do circuito A, a árvore de eventos para o evento topo “falta luz” é dada na Figura 1.2.10.

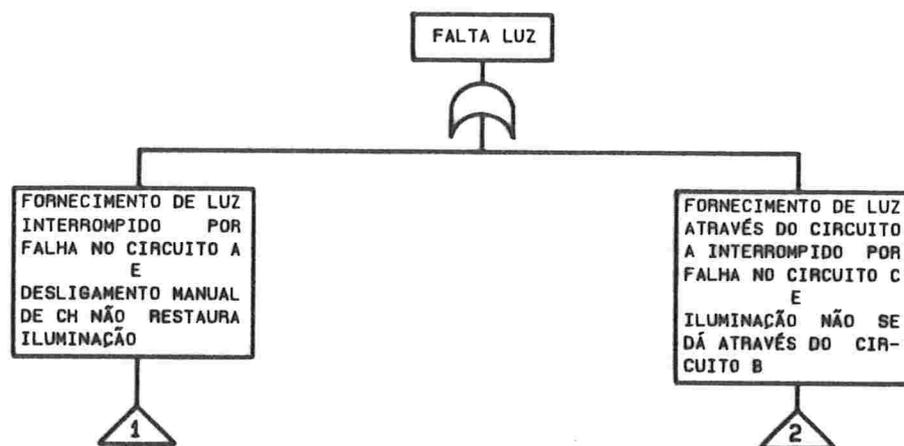


Figura 1.2.10: Árvore de eventos para o evento topo “falta luz”

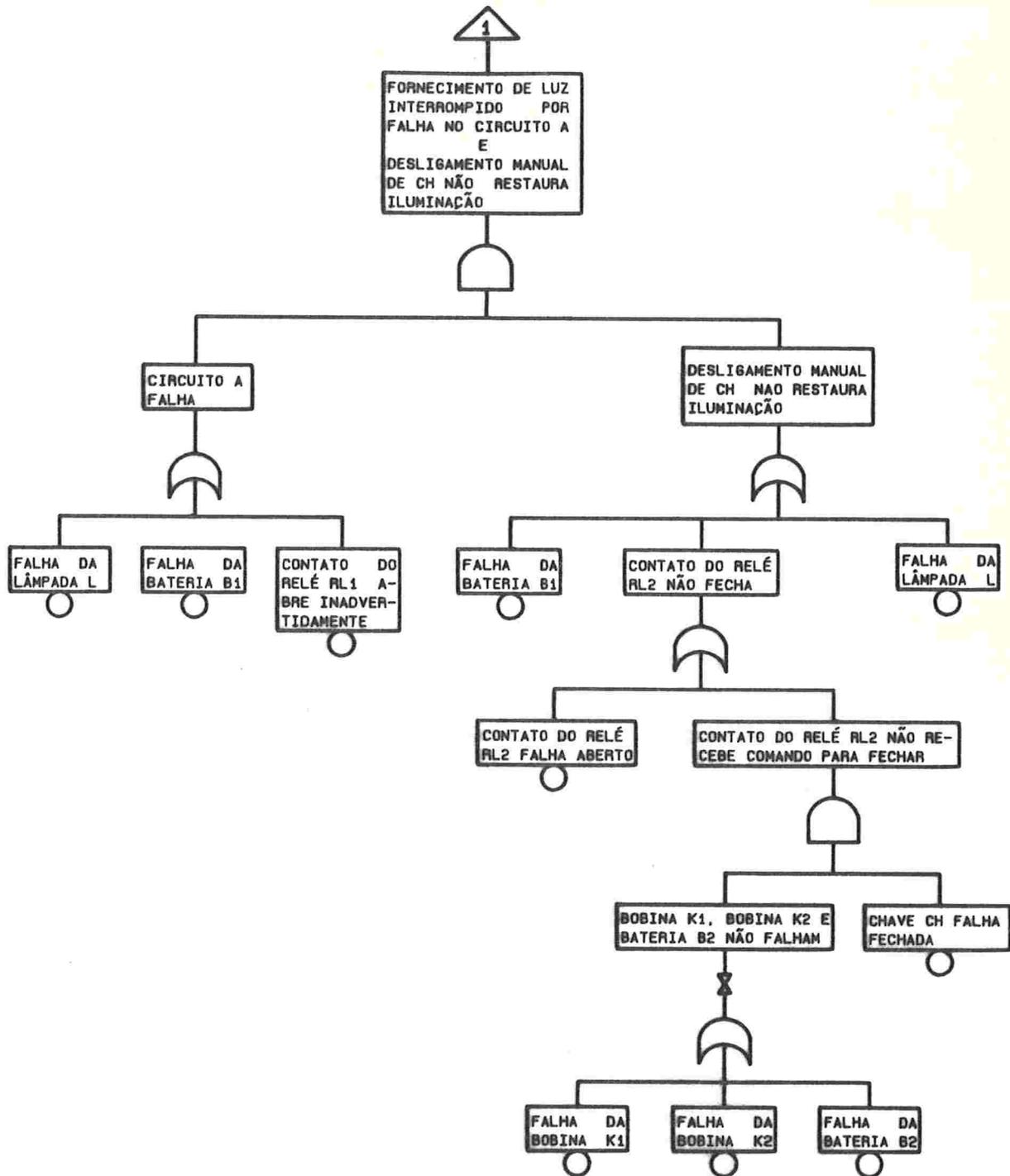


Figura 1.2.10: (continuação)

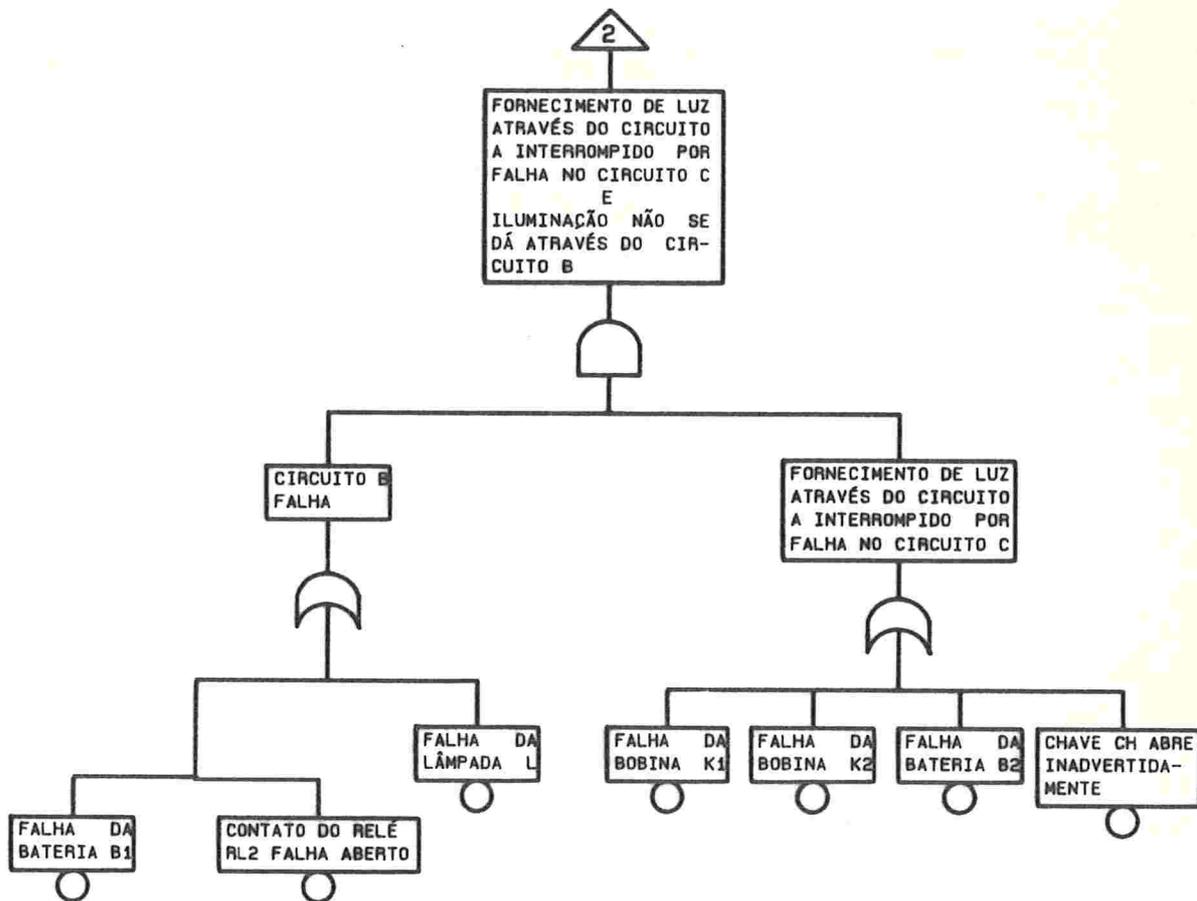
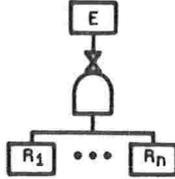
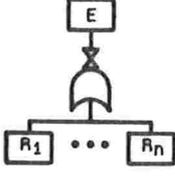
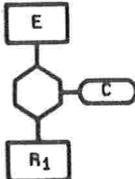
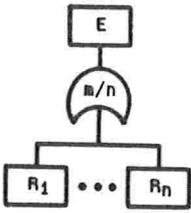
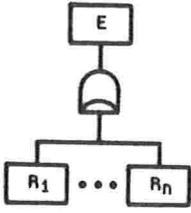


Figura 1.2.10: (continuação)

1.3 Portas Lógicas e Eventos Especiais

Apesar da possibilidade de se representar qualquer porta lógica através de uma combinação de portas lógicas fundamentais, em algumas situações, entretanto, portas lógicas especiais são utilizadas com o objetivo de representar de forma mais direta e simplificada algumas combinações lógicas mais complexas de eventos. A tabela 1.3.1 apresenta algumas dessas portas lógicas especiais, e outros exemplos podem ser encontrados em [Haa2].

Tabela 1.3.1: Símbolos de portas lógicas especiais

Nome da porta $\textcircled{1}$	Notação gráfica	Expressão local	Relação causal
“NAND ”		$E = (R_1 \cap \dots \cap R_n)^c$	o evento de saída E ocorre se pelo menos um dos eventos de entrada R_1, \dots, R_n não ocorrer
“NOR ”		$E = (R_1 \cup \dots \cup R_n)^c$	o evento de saída E ocorre se nenhum dos eventos de entrada R_1, \dots, R_n ocorrer
“INHIBIT”		$E = R_1 \cap C$	o evento de entrada R_1 produz o evento de saída E , quando a condição C , conectada à direita da porta lógica, está presente
“m/n”		$E = \bigcup_{1 \leq i_1 < \dots < i_m \leq n} \bigcap_{i \in \{i_1, \dots, i_m\}} R_i$	o evento de saída E ocorre, se m dos n eventos de entrada R_1, \dots, R_n ocorrerem simultaneamente
“EXCLUSIVE OR”		$E = \bigcup_{i=1}^n (R_i \cap (\bigcap_{j \neq i} R_j^c))$	o evento de saída E ocorre se um, e apenas um, dos eventos de entrada R_1, \dots, R_n ocorrerem

Cada expressão local correspondente à uma porta lógica especial pode ser transformada em uma combinação de expressões locais correspondentes à portas lógicas fundamentais, utilizando as identidades básicas da teoria dos conjuntos. Assim, denominando as expressões locais correspondentes à portas lógicas especiais de expressões locais especiais e as correspondentes à portas lógicas fundamentais de expressões locais fundamentais, as identidades para as expressões locais especiais contidas na Tabela 1.3.1 são dadas na Tabela 1.3.2 abaixo.

Tabela 1.3.2: Identidade entre expressões locais especiais e expressões locais fundamentais

Nome da porta \textcircled{f}	Expressão local especial	Expressão local fundamental
“NAND ”	$E = (R_1 \cap \dots \cap R_n)^c$	$E = R_1^c \cup \dots \cup R_n^c$
“NOR ”	$E = (R_1 \cup \dots \cup R_n)^c$	$E = R_1^c \cap \dots \cap R_n^c$
“INHIBIT”	$E = R_1 \cap C$	$E = R_1 \cap C$
“m/n”	$E = \bigcup_{1 \leq i_1 < \dots < i_m \leq n} \bigcap_{i \in \{i_1, \dots, i_m\}} R_i$	$E = \bigcup_{1 \leq i_1 < \dots < i_m \leq n} \bigcap_{i \in \{i_1, \dots, i_m\}} R_i$
“EXCLUSIVE OR”	$\bigcup_{i=1}^n (R_i \cap (\bigcap_{j \neq i} R_j^c))$	$\bigcup_{i=1}^n (R_i \cap (\bigcap_{j \neq i} R_j^c))$

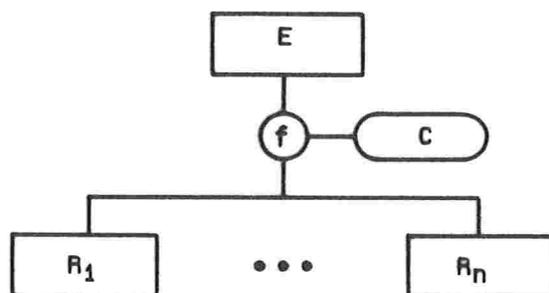
Além das portas lógicas especiais, existem situações onde deseja-se avaliar ou levar em consideração, os efeitos de uma condição bem definida. Nestes casos, dois tipos especiais de eventos terminais são utilizados, cujos nomes e notação gráfica estão apresentados na tabela 1.3.3.

Tabela 1.3.3: Eventos terminais especiais

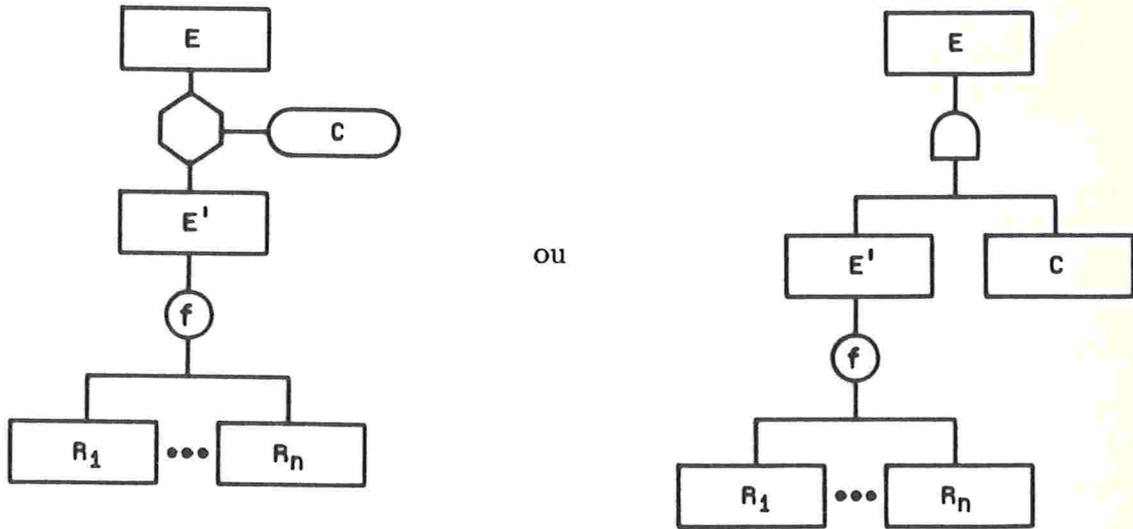
<i>Nome do evento</i>	<i>Notação gráfica</i>
evento terminal disparador	
evento terminal condicionador	

O evento terminal disparador (Tabela 1.3.3) age como uma chave. Quando a chave está ativa, a árvore admite como certa a ocorrência do evento indicado; caso contrário, a árvore admite como certa a sua não ocorrência. Esse tipo de evento pode também ser utilizado para anular o efeito dos eventos de entrada de uma porta lógica “AND” ou “OR”. Com efeito, desativando-se (ativando-se) um evento terminal disparador fictício colocado como um evento de entrada da porta lógica “AND” (“OR”), anula-se o efeito de todos os eventos de entrada dessa porta.

O evento terminal condicionador (Tabela 1.3.3), utilizado na porta lógica “INHIBIT”, pode também ser conectado a qualquer porta lógica para indicar que o evento de saída ocorre se, e somente se, o evento condicionador indicado ocorre. Assim,



indica que a ocorrência de E é equivalente a ocorrência simultânea de C e do evento resultante da aplicação de \odot à R_1, \dots, R_n . A mesma relação lógica pode ainda ser escrita como



que são graficamente mais complicadas.

Capítulo 2

Análise Qualitativa

2.1 Introdução

Durante a análise do projeto de um sistema complexo de engenharia, um analista de confiabilidade é colocado diante de perguntas tais como: Existem pontos fracos no projeto? O arranjo proposto para a instalação evita que fatores tais como explosões, incêndios ou inundações impliquem na perda total da capacidade operacional? Um outro “lay-out” evitaria tal perda de capacidade? O sistema é tolerante a falhas simples?

As atividades de projeto destinadas a responder tais perguntas constituem o que se denomina de análise qualitativa de confiabilidade. Nestas atividades, o analista recorre, fundamentalmente, à construção de modelos de árvore de eventos para o sistema em estudo, com o objetivo de sintetizar as relações mais elementares de causa e efeito entre múltiplas mudanças de estado de componentes e mudanças de estado do sistema. Esta síntese é feita através de classes especiais de eventos terminais básicos dos modelos desenvolvidos, cuja determinação é o objetivo deste capítulo. Nele serão introduzidos a conceituação matemática teórica do modelo de árvore de eventos e alguns algoritmos utilizados como ferramenta na determinação dessas classes.

2.2 Elementos Algébricos

Apresenta-se, nesta seção, a estrutura matemática formal para o método de análise de árvore de eventos. Para dar um tratamento qualitativo e quantitativo integrado, as árvores de eventos serão conceituadas no âmbito dos espaços probabilisáveis ou mensuráveis, e os métodos de análise serão abordados do ponto de vista da teoria dos conjuntos.

Seja (Ω, \mathcal{F}) um espaço probabilisável. Para simplificar a apresentação dos conceitos básicos da análise, se $B \subset \mathcal{F}$ é finito e não-vazio, utiliza-se a seguinte notação:

1. $\overline{B} \stackrel{\text{not}}{=} \{E^c : E \in B\}$
2. $\cup B \stackrel{\text{not}}{=} \cup_{E \in B} E$
3. $\cap B \stackrel{\text{not}}{=} \cap_{E \in B} E$
4. $B_o \stackrel{\text{not}}{=} B \cup \{\emptyset, \Omega\}$
5. $P^*(B) \stackrel{\text{not}}{=} \{A \in P(B) - \{\emptyset\} : A \cap \overline{A} = \emptyset\}$, onde $P(B)$ é a classe de todos os subconjuntos de B
6. $\sigma'(B) \stackrel{\text{not}}{=} \sigma(B) - \{\emptyset, \Omega\}$, onde $\sigma(B)$ é a σ -álgebra gerada por B .

Definição 2.2.1 Se $E \in \sigma'(B) - B$, onde $B \subset \mathcal{F}$ é finito, não-vazio e tal que $B \cap \overline{B} = \emptyset$, diz-se que E é um evento porta com entradas em B . Além disso, diz-se que $F \in B$ é entrada de E se $E \notin \sigma'(B - \{F\})$.

Definição 2.2.2 Diz-se que $\mathcal{A} = \{E_{ij} : 1 \leq i \leq m, 1 \leq j \leq n_i\} \subset \mathcal{F}$ é uma árvore de eventos, se existir $\mathcal{E}_{\mathcal{A}} \subset \mathcal{F}$ finito, não-vazio, tal que:

1. $\mathcal{E}_{\mathcal{A}} \cap \overline{\mathcal{E}_{\mathcal{A}}} = \emptyset$
2. $\emptyset, \Omega \notin \mathcal{E}_{\mathcal{A}}$

3. para $E, F \in (\mathcal{E}_A \cup \overline{\mathcal{E}_A})$ distintos, $E \not\subseteq F$
4. para $A \subset (\mathcal{E}_A \cup \overline{\mathcal{E}_A})$, $\cap A \neq \emptyset$ se, e somente se, $A \cap \overline{A} = \emptyset$
5. para $A, B \subseteq (\mathcal{E}_A \cup \overline{\mathcal{E}_A})$, $\cap A \subseteq \cap B$ se, e somente se, $A \supseteq B$
6. $n_1 = 1$ e E_{11} é um evento porta com entradas em $\{E_{2j} : 1 \leq j \leq n_2\}$
7. para $1 < i < m$, todo elemento de $\{E_{ij} : 1 \leq j \leq n_i\} - \mathcal{E}_A$ é um evento porta com entradas em $\{E_{(i+1)j} : 1 \leq j \leq n_{i+1}\}$ e todo elemento de $\{E_{(i+1)j} : 1 \leq j \leq n_{i+1}\}$ é entrada de algum elemento de $\{E_{ij} : 1 \leq j \leq n_i\} - \mathcal{E}_A$
8. $\{E_{mj} : 1 \leq j \leq n_m\} \subseteq \mathcal{E}_A$
9. $\mathcal{E}_A = \bigcup_{i=1}^m \{E_{ij} : 1 \leq j \leq n_i\} \cap \mathcal{E}_A$.

$E_{11} = T$ é denominado evento topo da árvore de eventos \mathcal{A} e os elementos de \mathcal{E}_A são denominados eventos terminais básicos de \mathcal{A} .

Para efeito de análise de confiabilidade, todos os eventos de $\overline{\mathcal{E}_A}$ são também considerados como eventos terminais básicos da árvore, apesar de aparecerem em \mathcal{A} sob a forma de eventos porta. Assim, será denotado por \mathcal{E} o conjunto $(\mathcal{E}_A \cup \overline{\mathcal{E}_A})$.

Algumas representações algébricas especiais são extremamente úteis na análise de árvore de eventos. Para introduzi-las, seja $\mathcal{E} \subset \mathcal{F}$ finito e não-vazio.

Definição 2.2.3 $\delta \subseteq \mathcal{P}^*(\sigma(\mathcal{E}))$ é uma disjunção para $E \in \sigma(\mathcal{E})$ se $\bigcup_{C \in \delta} \cap C = E$. A disjunção δ é normal se, para todo $C \in \delta$, a condição $C \in \mathcal{P}^*(\mathcal{E})$ estiver satisfeita. A disjunção δ é disjunta se, para quaisquer $A, B \in \delta$ distintos, a condição $A \cap \overline{B} \neq \emptyset$ estiver satisfeita.

Para apresentar algumas propriedades possuídas pelas disjunções, denote por \mathcal{D}° , o conjunto das disjunções que são subconjuntos de $\mathcal{P}^*(\sigma(\mathcal{E}))$ e, por \mathcal{D} , o conjunto das disjunções contidas em \mathcal{D}° , que são subconjuntos de $\mathcal{P}^*(\mathcal{E}_o)$.

Definição 2.2.4 Se δ e $\xi \in \mathcal{D}^\circ$ são disjunções para $E \in \sigma(\mathcal{E})$, diz-se que δ e ξ são equivalentes e escreve-se $\delta \equiv \xi$.

Definição 2.2.5 Se $\delta, \xi \in \mathcal{D}^\circ$, a soma de δ e ξ é definida como a disjunção

$$\delta \vee \xi = \delta \cup \xi.$$

Proposição 2.2.1 Se $\delta_i, \xi_i \in \mathcal{D}^\circ$ e $\delta_i \equiv \xi_i$, para $i = 1, \dots, k$, então $\bigvee_{i=1}^k \delta_i \equiv \bigvee_{i=1}^k \xi_i$.

□

Definição 2.2.6 Se $\delta, \xi \in \mathcal{D}^\circ$, o produto de δ e ξ é definido como a disjunção

$$\delta \wedge \xi = \begin{cases} \{A \cup B : A \cap \bar{B} = \emptyset, A \in \delta \text{ e } B \in \xi\} & \text{se o conjunto for não-vazio,} \\ \{\{\emptyset\}\} & \text{caso contrário.} \end{cases}$$

Proposição 2.2.2 Se δ, ξ e $\varphi \in \mathcal{D}^\circ$ e $\delta \equiv \varphi$, então $\delta \wedge \xi \equiv \varphi \wedge \xi$.

Prova - Como $\delta \equiv \varphi$, então $\bigcup_{A \in \delta} \bigcap A = \bigcup_{B \in \varphi} \bigcap B$, e tem-se

$$\begin{aligned} \bigcup_{D \in (\delta \wedge \xi)} \bigcap D &= \left(\bigcup_{A \in \delta} \bigcap A \right) \cap \left(\bigcup_{C \in \xi} \bigcap C \right) \\ &= \left(\bigcup_{B \in \varphi} \bigcap B \right) \cap \left(\bigcup_{C \in \xi} \bigcap C \right) \\ &= \bigcup_{E \in (\varphi \wedge \xi)} \bigcap E. \end{aligned}$$

Consequentemente, $\delta \wedge \xi \equiv \varphi \wedge \xi$.

□

Definição 2.2.7 $C \in \mathcal{P}^*(\mathcal{E}_\emptyset)$ é suficiente para $E \in \sigma(\mathcal{E})$ se $\bigcap C \subseteq E$. Neste caso, $\bigcap C$ é denominado evento suficiente para E . Se C é suficiente para E e $|C| = 1$ ou $|C| > 1$ e não existe $C' \subset C$ suficiente para E , então C é suficiente mínimo para E , e $\bigcap C$ é denominado evento suficiente mínimo para E .

Definição 2.2.8 $C \in \mathcal{P}^*(\mathcal{E}_o)$ é necessário para $E \in \sigma(\mathcal{E})$ se $\cup C \supseteq E$. Neste caso, $\cup C$ é denominado evento necessário para E . Se C é necessário para E e $|C| = 1$ ou $|C| > 1$ e não existe $C' \subset C$ necessário para E , então C é necessário mínimo para E , e $\cup C$ é denominado evento necessário mínimo para E .

Definição 2.2.9 $\varrho \subseteq \mathcal{P}^*(\sigma(\mathcal{E}))$ é uma conjunção para $E \in \sigma(\mathcal{E})$ se $\bigcap_{C \in \varrho} \cup C = E$. A conjunção ϱ é normal se, para todo $C \in \varrho$, a condição $C \in \mathcal{P}^*(\mathcal{E})$ estiver satisfeita.

Para $E \in \sigma(\mathcal{E})$, sejam

$$\mathcal{B}_E = \{C \in \mathcal{P}^*(\mathcal{E}_o) : C \text{ é suficiente mínimo para } E\}$$

e

$$\mathcal{R}_E = \{C \in \mathcal{P}^*(\mathcal{E}_o) : C \text{ é necessário mínimo para } E\}.$$

Então seguem as seguintes definições:

Definição 2.2.10 Uma disjunção $\delta \in \mathcal{D}$, para $E \in \sigma(\mathcal{E})$, é uma base para E se $\delta \subseteq \mathcal{B}_E$. A base δ é completa se $\delta = \mathcal{B}_E$. A base δ é irredundante se qualquer $\xi \subset \delta$ não for uma base para E . A base irredundante δ é mínima se $|\delta| \leq |\xi|$, para qualquer outra base irredundante ξ para E .

Definição 2.2.11 Uma conjunção $\varrho \subseteq \mathcal{P}^*(\mathcal{E}_o)$, para $E \in \sigma(\mathcal{E})$, é um recobrimento para E se $\varrho \subseteq \mathcal{R}_E$. O recobrimento ϱ é completo se $\varrho = \mathcal{R}_E$. O recobrimento ϱ é irredundante se qualquer $\varsigma \subset \varrho$ não for um recobrimento para E . O recobrimento irredundante ϱ é mínimo se $|\varrho| \leq |\varsigma|$, para qualquer outro recobrimento irredundante ς para E .

Definição 2.2.12 Sejam \mathcal{A} e \mathcal{A}^D duas árvores de eventos. \mathcal{A}^D é a árvore de eventos dual de \mathcal{A} , se $\mathcal{A}^D = \overline{\mathcal{A}}$.

Observações :

- Sempre que for mencionado evento topo T neste trabalho, $T \in \sigma'(\mathcal{E})$.

- Se ψ é uma disjunção (conjunção) para o evento topo T da árvore de eventos \mathcal{A} , então $\chi = \{\overline{C} : C \in \psi\}$ é uma conjunção (disjunção) para o evento topo T^c de \mathcal{A}^D , a árvore de eventos dual de \mathcal{A} .
- Se $\psi \subseteq \mathcal{P}^*(\mathcal{E}_o)$ é uma disjunção (conjunção) para $E \in \sigma(\mathcal{E})$, então todo elemento de ψ é suficiente (necessário) para E .
- Se C é suficiente (necessário) para o evento topo T da árvore de eventos \mathcal{A} , então \overline{C} é necessário (suficiente) para o evento topo T^c de \mathcal{A}^D , a árvore de eventos dual de \mathcal{A} .
- $E \in \sigma(\mathcal{E})$ possui uma única base completa, porém pode possuir mais de uma base irredundante. Considerações análogas valem para recobrimento.
- Se a base completa \mathcal{B} para o evento topo de uma árvore de eventos satisfaz $(\cup \mathcal{B}) \cap (\cup \overline{\mathcal{B}}) = \emptyset$, então a árvore é denominada monótona; caso contrário a árvore é denominada não-monótona. Se uma árvore de eventos é monótona, então: a) o evento topo T possui uma única base e um único recobrimento, que são ao mesmo tempo, completos e irredundantes; e b) os conjuntos suficientes mínimos são denominados cortes mínimos e os conjuntos necessários mínimos são denominados caminhos mínimos.

2.3 Base Completa

Algoritmos para a obtenção de bases completas foram objeto de estudo de vários pesquisadores (cf.: [HR], [HW], [KH], [Nel], [Reu], [Qui1]) interessados em análise de árvore de eventos, automatos finitos ou circuitos lógicos. Nesta seção, são apresentados alguns destes algoritmos, precedidos de uma introdução sobre a obtenção de disjunções e conjunções normais, já que os primeiros, em geral, partem de representações deste tipo. O contexto utilizado para a apresentação dos algoritmos será o de uma árvore de eventos \mathcal{A} , com conjunto de eventos terminais básicos \mathcal{E} e evento topo T , com a notação introduzida na seção 2.2.

2.3.1 Algoritmos para a Obtenção de Disjunções e Conjunções Normais

Os algoritmos comumente empregados para a obtenção de disjunções e conjunções normais para eventos topo de árvores de eventos, podem ser divididos em duas categorias:

1. algoritmos que utilizam expressões algébricas de partida para o evento topo em função dos eventos terminais básicos;
2. algoritmos que utilizam a própria estrutura lógica da árvore.

Para a obtenção de disjunções e conjunções normais, dois métodos extremamente simples, denominados método “top-down” e método “bottom-up”, são utilizados. Como os próprios nomes sugerem, o método “top-down” percorre a estrutura lógica da árvore partindo do evento topo, até atingir os eventos terminais básicos do último nível da árvore, e o método “bottom-up”, parte dos eventos terminais básicos do último nível da árvore, e percorre sua estrutura lógica, até atingir o evento topo.

Na obtenção de expressões algébricas de partida para os algoritmos da categoria 1., os métodos “top-down” e “bottom-up” são assim utilizados:

Método “Top-Down”: Começando com o evento topo da árvore de eventos, substitui-se sucessivamente cada evento porta por sua expressão local, conforme a fórmula que o define em função de suas entradas, até obter uma representação para o evento topo envolvendo apenas eventos terminais básicos.

Método “Bottom-Up”: Começando com os eventos porta cujas entradas são eventos terminais básicos, substitui-se sucessivamente estes eventos por suas expressões locais, conforme a fórmula que os definem em função de suas entradas, até que o evento topo da árvore de eventos seja substituído.

Para os algoritmos da categoria 2., os métodos “top-down” e “bottom-up” são utilizados de forma específica, de acordo com a lógica envolvida em cada algoritmo.

Exemplo 2.3.1 Para exemplificar a aplicação dos algoritmos para a obtenção de uma disjunção/conjunção normal, será utilizada a árvore de eventos para o evento topo “falta luz”, referente ao problema do sistema de iluminação descrito no Exemplo 1.2.3, que é dada na Figura 1.2.10 (Capítulo 1). A definição dos eventos da árvore é dada a seguir.

- $$E_{11} = \text{“falta luz”},$$
- $$E_{21} = \text{“fornecimento de luz interrompido por falha no circuito A e desligamento manual de CH não restaura iluminação”}$$
- $$E_{22} = \text{“fornecimento de luz através do circuito A interrompido por falha no circuito C e iluminação não se dá através do circuito B ”}$$
- $$E_{31} = \text{“circuito A falha”}$$
- $$E_{32} = \text{“desligamento manual de CH não restaura iluminação”}$$
- $$E_{33} = \text{“circuito B falha”}$$
- $$E_{34} = \text{“fornecimento de luz através do circuito A interrompido por falha no circuito C”}$$
- $$E_{41} = \text{“falha da lâmpada L”}$$
- $$= L$$
- $$E_{42} = \text{“falha da bateria B1”}$$
- $$= B1$$
- $$E_{43} = \text{“contato do relé RL1 abre inadvertidamente”}$$
- $$= RL1_{ai}$$
- $$E_{44} = B1$$
- $$E_{45} = \text{“contato do relé RL2 não fecha”}$$
- $$E_{46} = L$$
- $$E_{47} = B1$$
- $$E_{48} = \text{“contato do relé RL2 falha aberto”}$$
- $$= RL2_{fa}$$
- $$E_{49} = L$$
- $$E_{4(10)} = \text{“falha da bobina K1”}$$
- $$= K1$$

$$\begin{aligned}
 E_{4(11)} &= \text{“falha da bobina K2”} \\
 &= K2 \\
 E_{4(12)} &= \text{“falha da bateria B2”} \\
 &= B2 \\
 E_{4(13)} &= \text{“chave CH abre inadvertidamente”} \\
 &= CH_{ai} \\
 E_{51} &= RL2_{fa} \\
 E_{52} &= \text{“contato do relé RL2 não recebe comando para fechar”} \\
 E_{61} &= \text{“bobina K1 , bobina K2 e bateria B2 não falham”} \\
 E_{62} &= \text{“chave CH falha fechada”} \\
 &= CH_{ff} \\
 E_{71} &= K1 \\
 E_{72} &= K2 \\
 E_{73} &= B2.
 \end{aligned}$$

Algoritmo DN

O algoritmo DN é um método para a obtenção de disjunções normais, que opera sobre expressões algébricas, e é executado através das seguintes etapas:

- DN.1 caso não se disponha de uma expressão algébrica para o evento topo de interesse, derive uma expressão, através dos métodos “top-down” ou “bottom-up”, substituindo cada evento porta por sua expressão local correspondente. Se a expressão local para um evento porta for especial, transforme-a em uma combinação de expressões locais fundamentais, logo após a substituição do evento porta por sua expressão local (veja Tabelas 1.2.1, 1.3.1 e 1.3.2 - Capítulo 1); caso contrário, vá para a etapa seguinte;
- DN.2 transforme a expressão algébrica disponível para o evento topo de interesse em uma expressão da forma $\bigcup_{C \in \delta} \bigcap C$, onde δ é uma disjunção normal. Para realizar esta transformação, utilize um processo de aplicação das leis distributivas e das leis de De

Morgan (caso necessário), complementado por simplificações do tipo:

$$\begin{cases} A \cap B = \emptyset & \text{se } A = B^c, \\ A \cap A = A, \\ A \cup (A \cap B) = A, \end{cases} \quad (2.3.1)$$

onde A e $B \in \sigma'(\mathcal{E})$.

Cada conjunto formado pelos eventos terminais básicos que compõem cada intersecção de eventos, da união de eventos obtida no final deste processo, é um elemento de δ , e a classe cujos elementos são estes conjuntos, corresponde a disjunção normal δ .

Observação:

- Este algoritmo encontra-se implementado no programa SETS ([Wor1], [Wor2], [WS]), utilizando o método “top-down”.

O programa SETS permite a utilização dos eventos porta “AND”, “OR”, “EXCLUSIVE OR” e de qualquer outro evento porta cuja expressão local seja especificada nos dados de entrada do programa.

Exemplo 2.3.2 Utilizando os eventos definidos no Exemplo 2.3.1, a disjunção normal para o evento topo “falta luz”, obtida através da aplicação do algoritmo DN é dada por:

Etapa DN.1 Através do método “bottom-up”:

$$\begin{aligned} E_{61} &= (E_{71} \cup E_{72} \cup E_{73})^c \\ &= (K1 \cup K2 \cup B2)^c \\ E_{62} &= CH_{ff} \\ E_{52} &= E_{61} \cap E_{62} \\ &= (K1 \cup K2 \cup B2)^c \cap CH_{ff} \\ E_{51} &= RL2_{fa} \\ E_{46} &= L \\ E_{45} &= E_{51} \cup E_{52} \\ &= RL2_{fa} \cup ((K1 \cup K2 \cup B2)^c \cap CH_{ff}) \end{aligned}$$

$$E_{44} = B1$$

$$\begin{aligned} E_{32} &= E_{44} \cup E_{45} \cup E_{46} \\ &= B1 \cup L \cup \left(RL2_{fa} \cup \left((K1 \cup K2 \cup B2)^c \cap CH_{ff} \right) \right) \end{aligned}$$

$$E_{43} = RL1_{ai}$$

$$E_{42} = B1$$

$$E_{41} = L$$

$$\begin{aligned} E_{31} &= E_{41} \cup E_{42} \cup E_{43} \\ &= L \cup B1 \cup RL1_{ai} \end{aligned}$$

$$\begin{aligned} E_{21} &= E_{31} \cap E_{32} \\ &= (L \cup B1 \cup RL1_{ai}) \cap \left(B1 \cup L \cup \left(RL2_{fa} \cup \left((K1 \cup K2 \cup B2)^c \cap CH_{ff} \right) \right) \right) \end{aligned}$$

$$E_{4(13)} = CH_{ai}$$

$$E_{4(12)} = B2$$

$$E_{4(11)} = K2$$

$$E_{4(10)} = K1$$

$$\begin{aligned} E_{34} &= E_{4(10)} \cup E_{4(11)} \cup E_{4(12)} \cup E_{4(13)} \\ &= K1 \cup K2 \cup B2 \cup CH_{ai} \end{aligned}$$

$$E_{49} = L$$

$$E_{48} = RL2_{fa}$$

$$E_{47} = B1$$

$$\begin{aligned} E_{33} &= E_{47} \cup E_{48} \cup E_{49} \\ &= B1 \cup RL2_{fa} \cup L \end{aligned}$$

$$\begin{aligned} E_{22} &= E_{33} \cap E_{34} \\ &= (B1 \cup RL2_{fa} \cup L) \cap (K1 \cup K2 \cup B2 \cup CH_{ai}) \end{aligned}$$

$$\begin{aligned} E_{11} &= E_{21} \cup E_{22} \\ &= \left((L \cup B1 \cup RL1_{ai}) \cap \left(B1 \cup L \cup \left(RL2_{fa} \cup \left((K1 \cup K2 \cup B2)^c \cap CH_{ff} \right) \right) \right) \right) \cup \\ &\quad \left((B1 \cup RL2_{fa} \cup L) \cap (K1 \cup K2 \cup B2 \cup CH_{ai}) \right) \end{aligned}$$

$$T = E_{11}.$$

Etapa DN.2 Pela aplicação das leis distributivas e das leis de De Morgan:

$$\begin{aligned}
T = & (L \cap B1) \cup (L \cap RL2_{fa}) \cup (L \cap K1^c \cap K2^c \cap B2^c \cap CH_{ff}) \cup \\
& (L \cap L) \cup (B1 \cap B1) \cup (B1 \cap RL2_{fa}) \cup \\
& (B1 \cap K1^c \cap K2^c \cap B2^c \cap CH_{ff}) \cup (B1 \cap L) \cup (RL1_{fa} \cap B1) \cup \\
& (RL1_{ai} \cap RL2_{fa}) \cup (RL1_{ai} \cap K1^c \cap K2^c \cap B2^c \cap CH_{ff}) \cup (RL1_{ai} \cap L) \cup \\
& (B1 \cap K1) \cup (B1 \cap K2) \cup (B1 \cap B2) \cup \\
& (B1 \cap CH_{ai}) \cup (RL2_{fa} \cap K1) \cup (RL2_{fa} \cap K2) \cup \\
& (RL2_{fa} \cap B2) \cup (RL2_{fa} \cap CH_{ai}) \cup (L \cap K1) \cup \\
& (L \cap K2) \cup (L \cap B2) \cup (L \cap CH_{ai}).
\end{aligned}$$

Após as simplificações dadas em (2.3.1):

$$\begin{aligned}
T = & L \cup B1 \cup (RL1_{ai} \cap RL2_{fa}) \cup (RL1_{ai} \cap K1^c \cap K2^c \cap B2^c \cap CH_{ff}) \cup \\
& (RL2_{fa} \cap K1) \cup (RL2_{fa} \cap K2) \cup (RL2_{fa} \cap B2) \cup (RL2_{fa} \cap CH_{ai}).
\end{aligned}$$

Logo,

$$\begin{aligned}
\delta = & \{ \{L\}, \{B1\}, \{RL1_{ai}, RL2_{fa}\}, \{RL1_{ai}, K1^c, K2^c, B2^c, CH_{ff}\}, \\
& \{RL2_{fa}, K1\}, \{RL2_{fa}, K2\}, \{RL2_{fa}, B2\}, \{RL2_{fa}, CH_{ai}\} \},
\end{aligned}$$

é a disjunção normal obtida no final do algoritmo DN.

Algoritmo de Fussell, Henry & Marshall

O algoritmo de Fussell, Henry & Marshall [FHM] é um método “top-down” para a obtenção de disjunções normais, que opera diretamente sobre a árvore de eventos. Originalmente desenvolvido para árvores construídas utilizando apenas as portas lógicas fundamentais “AND” e “OR” (Programa MOCUS [FHM]), este algoritmo foi posteriormente modificado para permitir a utilização das portas lógicas “AND”, “OR”, “NOT”, “NAND” e “NOR” (Programa MOCUS-M [Vie]).

A execução deste algoritmo se dá através das seguintes etapas:

- FHM.1 utilizando a árvore de eventos para o evento topo de interesse, monte uma matriz, colocando o evento topo como elemento da primeira linha e primeira coluna da matriz;
- FHM.2 substitua cada evento porta “OR” (“NAND” ou (“NOT” e “AND”)) da árvore, por suas entradas (pelos complementares de suas entradas), criando assim novas linhas na matriz que são idênticas, a menos da posição antes ocupada pelo evento porta “OR” (“NAND” ou (“NOT” e “AND”)) que torna-se ocupada por suas entradas (pelos complementares de suas entradas); substitua cada evento porta “AND” (“NOR” ou (“NOT” e “OR”)) da árvore, por suas entradas (pelos complementares de suas entradas), colocando-os na mesma linha da matriz montada, aumentando assim o número de colunas da matriz;
- FHM.3 começando da primeira coluna da matriz, repita FHM.2 até que todos os elementos da coluna considerada sejam eventos terminais básicos;
- FHM.4 na matriz obtida em FHM.3: elimine nas linhas, as repetições de eventos terminais básicos; elimine as linhas contendo eventos terminais básicos complementares e as que incluem todos os elementos de uma outra linha qualquer da matriz.

δ , a classe formada pelos conjuntos cujos elementos são os elementos de cada linha da matriz obtida no final deste processo, é uma disjunção normal para o evento topo de interesse.

Exemplo 2.3.3 Utilizando os eventos definidos no Exemplo 2.3.1, a disjunção normal para o evento topo “falta luz”, obtida através da aplicação do algoritmo de Fussell, Henry & Marshall é dada por:

Etapas FHM.1 a FHM.3

$$\begin{array}{l}
 1. \left(E_{11} \right) \qquad 2. \begin{pmatrix} E_{21} \\ E_{22} \end{pmatrix} \\
 3. \begin{pmatrix} E_{31} & E_{32} \\ E_{22} & \end{pmatrix} \qquad 4. \begin{pmatrix} L & E_{32} \\ B1 & E_{32} \\ RL1_{ai} & E_{32} \\ E_{22} & \end{pmatrix} \qquad 5. \begin{pmatrix} L & E_{32} \\ B1 & E_{32} \\ RL1_{ai} & E_{32} \\ E_{33} & E_{34} \end{pmatrix}
 \end{array}$$

$$6. \begin{pmatrix} L & E_{32} \\ B1 & E_{32} \\ RL1_{ai} & E_{32} \\ B1 & E_{34} \\ RL2_{fa} & E_{34} \\ L & E_{34} \end{pmatrix}$$

$$7. \begin{pmatrix} L & B1 \\ L & E_{45} \\ L & L \\ B1 & B1 \\ B1 & E_{45} \\ B1 & L \\ RL1_{ai} & B1 \\ RL1_{ai} & E_{45} \\ RL1_{ai} & L \\ B1 & E_{34} \\ RL2_{fa} & E_{34} \\ L & E_{34} \end{pmatrix}$$

$$8. \begin{pmatrix} L & B1 \\ L & RL2_{fa} \\ L & E_{52} \\ L & L \\ B1 & B1 \\ B1 & RL2_{fa} \\ B1 & E_{52} \\ B1 & L \\ RL1_{ai} & B1 \\ RL1_{ai} & RL2_{fa} \\ RL1_{ai} & E_{52} \\ RL1_{ai} & L \\ B1 & E_{34} \\ RL2_{fa} & E_{34} \\ L & E_{34} \end{pmatrix}$$

$$9. \begin{pmatrix} L & B1 \\ L & RL2_{fa} \\ L & E_{61} & CH_{ff} \\ L & L \\ B1 & B1 \\ B1 & RL2_{fa} \\ B1 & E_{61} & CH_{ff} \\ B1 & L \\ RL1_{ai} & B1 \\ RL1_{ai} & RL2_{fa} \\ RL1_{ai} & E_{61} & CH_{ff} \\ RL1_{ai} & L \\ B1 & E_{34} \\ RL2_{fa} & E_{34} \\ L & E_{34} \end{pmatrix}$$

$$10. \begin{pmatrix} L & B1 \\ L & RL2_{fa} \\ L & K1^c & K2^c & B2^c & CH_{ff} \\ L & L \\ B1 & B1 \\ B1 & RL2_{fa} \\ B1 & K1^c & K2^c & B2^c & CH_{ff} \\ B1 & L \\ RL1_{ai} & B1 \\ RL1_{ai} & RL2_{fa} \\ RL1_{ai} & K1^c & K2^c & B2^c & CH_{ff} \\ RL1_{ai} & L \\ B1 & E_{34} \\ RL2_{fa} & E_{34} \\ L & E_{34} \end{pmatrix}$$

$$11. \left(\begin{array}{l} L \quad B1 \\ L \quad RL2_{fa} \\ L \quad K1^c \quad K2^c \quad B2^c \quad CH_{ff} \\ L \quad L \\ B1 \quad B1 \\ B1 \quad RL2_{fa} \\ B1 \quad K1^c \quad K2^c \quad B2^c \quad CH_{ff} \\ B1 \quad L \\ RL1_{ai} \quad B1 \\ RL1_{ai} \quad RL2_{fa} \\ RL1_{ai} \quad K1^c \quad K2^c \quad B2^c \quad CH_{ff} \\ RL1_{ai} \quad L \\ B1 \quad K1 \\ B1 \quad K2 \\ B1 \quad B2 \\ B1 \quad CH_{ai} \\ RL2_{fa} \quad K1 \\ RL2_{fa} \quad K2 \\ RL2_{fa} \quad B2 \\ RL2_{fa} \quad CH_{ai} \\ L \quad K1 \\ L \quad K2 \\ L \quad B2 \\ L \quad CH_{ai} \end{array} \right)$$

Etapa FHM.4

$$\left(\begin{array}{l} L \\ B1 \\ RL1_{ai} \quad RL2_{fa} \\ RL1_{ai} \quad K1^c \quad K2^c \quad B2^c \quad CH_{ff} \\ RL2_{fa} \quad K1 \\ RL2_{fa} \quad K2 \\ RL2_{fa} \quad B2 \\ RL2_{fa} \quad CH_{ai} \end{array} \right)$$

Logo,

$$\delta = \left\{ \{L\}, \{B1\}, \{RL1_{ai}, RL2_{fa}\}, \{RL1_{ai}, K1^c, K2^c, B2^c, CH_{ff}\}, \right. \\ \left. \{RL2_{fa}, K1\}, \{RL2_{fa}, K2\}, \{RL2_{fa}, B2\}, \{RL2_{fa}, CH_{ai}\} \right\},$$

é a disjunção normal obtida através do algoritmo de Fussell, Henry & Marshall.

Algoritmo CN

O algoritmo CN é um método para a obtenção de conjunções normais, que opera sobre expressões algébricas, e é executado através das seguintes etapas:

- CN.1 caso não se disponha de uma expressão algébrica para o evento topo de interesse, derive uma expressão, através dos métodos “top-down” ou “bottom-up”, substituindo cada evento porta por sua expressão local correspondente. Se a expressão local para um evento porta for especial, transforme-a em uma combinação de expressões locais fundamentais, logo após a substituição do evento porta por sua expressão local (veja Tabelas 1.2.1, 1.3.1 e 1.3.2 - Capítulo 1); caso contrário, vá para a etapa seguinte;
- CN.2 complemente a expressão disponível para o evento topo de interesse e transforme-a em uma expressão da forma $\bigcup_{C \in \delta} C$, onde δ é uma disjunção normal para o complementar do evento topo de interesse. Para realizar esta transformação, utilize um processo de aplicação das leis distributivas e das leis de De Morgan, complementado por simplificações do tipo:

$$\begin{cases} A \cap B = \emptyset & \text{se } A = B^c, \\ A \cap A = A, \\ A \cup (A \cap B) = A, \end{cases}$$

onde A e $B \in \sigma'(\mathcal{E})$.

- CN.3 complemente a expressão obtida em CN.2 e aplique as leis de De Morgan, obtendo assim uma expressão da forma $\bigcap_{C \in \varrho} C$, onde ϱ é uma conjunção normal para o evento topo de interesse.

De forma análoga a descrita no algoritmo DN, cada conjunto formado pelos eventos terminais básicos que compõem cada união de eventos, da intersecção de eventos obtida no final deste processo, é um elemento de ϱ , e a classe cujos elementos são estes conjuntos, corresponde a ϱ .

Exemplo 2.3.4 Para exemplificar a obtenção de uma conjunção normal através deste algoritmo, considere a expressão resultante da aplicação do algoritmo DN ao evento topo T = “falta luz” (Exemplo 2.3.2).

Etapa CN.2

$$T^c = (L \cup B1 \cup (RL1_{ai} \cap RL2_{fa}) \cup (RL1_{ai} \cap K1^c \cap K2^c \cap B2^c \cap CH_{ff}) \cup (RL2_{fa} \cap K1) \cup (RL2_{fa} \cap K2) \cup (RL2_{fa} \cap B2) \cup (RL2_{fa} \cap CH_{ai}))^c.$$

$$T^c = L^c \cap B1^c \cap (RL1_{ai}^c \cup RL2_{fa}^c) \cap (RL1_{ai}^c \cup K1 \cup K2 \cup B2 \cup CH_{ff}^c) \cap (RL2_{fa}^c \cup K1^c) \cap (RL2_{fa}^c \cup K2^c) \cap (RL2_{fa}^c \cup B2^c) \cap (RL2_{fa}^c \cup CH_{ai}^c).$$

$$T^c = ((L^c \cap B1^c \cap RL1_{ai}^c) \cup (L^c \cap B1^c \cap RL2_{fa}^c)) \cap ((RL1_{ai}^c \cup K1 \cup K2 \cup B2 \cup CH_{ff}^c) \cap (RL2_{fa}^c \cup K1^c) \cap (RL2_{fa}^c \cup K2^c) \cap (RL2_{fa}^c \cup B2^c) \cap (RL2_{fa}^c \cup CH_{ai}^c)).$$

$$T^c = ((L^c \cap B1^c \cap RL1_{ai}^c) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap K1) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap K2) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap B2) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap CH_{ff}^c)) \cap ((RL2_{fa}^c \cup K1^c) \cap (RL2_{fa}^c \cup K2^c) \cap (RL2_{fa}^c \cup B2^c) \cap (RL2_{fa}^c \cup CH_{ai}^c)).$$

$$T^c = ((L^c \cap B1^c \cap RL1_{ai}^c \cap RL2_{fa}^c) \cup (L^c \cap B1^c \cap RL1_{ai}^c \cap K1^c) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap K1) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap K2) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap B2) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap CH_{ff}^c)) \cap ((RL2_{fa}^c \cup K2^c) \cap (RL2_{fa}^c \cup B2^c) \cap (RL2_{fa}^c \cup CH_{ai}^c)).$$

$$T^c = \left((L^c \cap B1^c \cap RL1_{ai}^c \cap RL2_{fa}^c) \cup (L^c \cap B1^c \cap RL1_{ai}^c \cap K1^c \cap K2^c) \cup \right. \\ \left. (L^c \cap B1^c \cap RL2_{fa}^c \cap K1) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap K2) \cup \right. \\ \left. (L^c \cap B1^c \cap RL2_{fa}^c \cap B2) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap CH_{ff}^c) \right) \cap \\ \left((RL2_{fa}^c \cup B2^c) \cap (RL2_{fa}^c \cup CH_{ai}^c) \right).$$

$$T^c = \left((L^c \cap B1^c \cap RL1_{ai}^c \cap RL2_{fa}^c) \cup (L^c \cap B1^c \cap RL1_{ai}^c \cap K1^c \cap K2^c \cap B2^c) \cup \right. \\ \left. (L^c \cap B1^c \cap RL2_{fa}^c \cap K1) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap K2) \cup \right. \\ \left. (L^c \cap B1^c \cap RL2_{fa}^c \cap B2) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap CH_{ff}^c) \right) \cap \\ (RL2_{fa}^c \cup CH_{ai}^c).$$

$$T^c = (L^c \cap B1^c \cap RL1_{ai}^c \cap RL2_{fa}^c) \cup (L^c \cap B1^c \cap RL1_{ai}^c \cap K1^c \cap K2^c \cap B2^c \cap CH_{ai}^c) \cup \\ (L^c \cap B1^c \cap RL2_{fa}^c \cap K1) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap K2) \cup \\ (L^c \cap B1^c \cap RL2_{fa}^c \cap B2) \cup (L^c \cap B1^c \cap RL2_{fa}^c \cap CH_{ff}^c).$$

Etapa CN.3

$$T = (L \cup B1 \cup RL1_{ai} \cup RL2_{fa}) \cap (L \cup B1 \cup RL1_{ai} \cup K1 \cup K2 \cup B2 \cup CH_{ai}) \cap \\ (L \cup B1 \cup RL2_{fa} \cup K1^c) \cap (L \cup B1 \cup RL2_{fa} \cup K2^c) \cap \\ (L \cup B1 \cup RL2_{fa} \cup B2^c) \cap (L \cup B1 \cup RL2_{fa} \cap CH_{ff}).$$

Logo,

$$\varrho = \left\{ \{L, B1, RL1_{ai}, RL2_{fa}\}, \{L, B1, RL1_{ai}, K1, K2, B2, CH_{ai}\}, \right. \\ \left. \{L, B1, RL2_{fa}, K1^c\}, \{L, B1, RL2_{fa}, K2^c\}, \right. \\ \left. \{L, B1, RL2_{fa}, B2^c\}, \{L, B1, RL2_{fa}, CH_{ff}\} \right\}.$$

Observação:

- Também pode-se obter uma conjunção normal através da aplicação do algoritmo de Fussell, Henry & Marshall à árvore de eventos dual para o evento topo de interesse. Especificamente, a classe constituída por cada conjunto formado pelos complementares dos elementos de cada linha da matriz obtida no final deste procedimento, é uma conjunção normal para o evento topo de interesse.

2.3.2 Algoritmos para a Obtenção da Base Completa

Se um dos algoritmos para a obtenção de uma disjunção normal foi aplicado à uma árvore de eventos \mathcal{A} e obtêve-se uma disjunção normal δ para o evento topo T de \mathcal{A} satisfazendo $(\cup\delta) \cap (\overline{\cup\delta}) = \emptyset$, então δ é a base completa para T . Especificamente, cada elemento da disjunção normal obtida é um conjunto suficiente mínimo para T e, reciprocamente, todo conjunto suficiente mínimo para T pertence a disjunção normal obtida [JM].

Em geral, porém, em árvores de eventos \mathcal{A} construídas utilizando as portas lógicas “NOT”, “NAND”, “NOR” ou “EXCLUSIVE OR”, a aplicação do mesmo procedimento fornece apenas uma disjunção normal δ para T . Para entender o que acontece, suponha, por exemplo, que aplicou-se o algoritmo DN à uma árvore de eventos \mathcal{A} , obtendo a expressão $T = (A \cap B) \cup (A^c \cap C)$ para o evento topo T de \mathcal{A} . Não é difícil ver que $\{A, B\}$ e $\{A^c, C\}$ são suficientes mínimos para T e que $\{B, C\}$, que não pertence a disjunção normal obtida, também é suficiente mínimo para T . Portanto, nesses casos, algoritmos mais específicos são necessários para a obtenção da base completa para T , já que a simples obtenção de uma disjunção normal, do tipo fornecido pelos algoritmos dados na Seção 2.3.1, não garante que todos os conjuntos suficientes mínimos para T tenham sido gerados.

Apresenta-se a seguir, os algoritmos propostos por Quine [Qui1] e Nelson [Nel], que são clássicos para a obtenção da base completa para um evento topo T , cuja árvore de eventos \mathcal{A} , se enquadra na situação exposta no parágrafo anterior. Por último, tentando melhorar sua linguagem, será apresentado o algoritmo proposto por Kumamoto & Henley [KH], que foi motivo de muita polêmica (cf. [LOKH], [WSH]) devido a dificuldade na sua compreensão, principalmente, por causa da notação rebuscada utilizada.

Exemplo 2.3.5 Para ilustrar a aplicação dos algoritmos a serem apresentados nesta seção, será utilizada a árvore de eventos do Exemplo 1 de Ogunbiyi, contido em [LOKH], já que a estrutura da mesma permite evidenciar de forma mais clara as operações envolvidas em cada um deles. Especificamente, a árvore é definida por

$$\mathcal{A} = \{E_{ij} : 1 \leq i \leq 4, 1 \leq j \leq n_i\},$$

onde

1. $\mathcal{E}_{\mathcal{A}} = \{P, Q, R, S\}$,
2. $\mathcal{E} = \{P, Q, R, S, P^c, Q^c, R^c, S^c\}$,
3. $n_2 = 2$, $n_3 = 5$ e $n_4 = 3$,

e as expressões locais dos eventos de \mathcal{A} são dadas por:

$$E_{11} = E_{21} \cup E_{22}$$

$$E_{21} = Q \cap E_{32}$$

$$E_{22} = Q^c \cap R \cap S^c$$

$$E_{31} = Q$$

$$E_{32} = P \cup S \cup R^c$$

$$E_{33} = Q^c$$

$$E_{34} = R$$

$$E_{35} = S^c$$

$$E_{41} = P$$

$$E_{42} = S$$

$$E_{43} = R^c.$$

Algoritmo de Quine

Quine [Qui1] propôs um algoritmo para a obtenção da base completa para um evento topo T , de uma árvore de eventos \mathcal{A} , partindo de uma disjunção normal δ para T . Este algoritmo, conhecido como método do consenso, é o método mais simples para a obtenção da base completa do evento topo de uma árvore de eventos cuja estrutura não é complexa.

Definição 2.3.1 *Sejam $A, B \in \mathcal{P}^*(\mathcal{E})$ tais que $A \cap \bar{B} = \{E\}$, $A \neq \{E\}$ e $B \neq \{E^c\}$. Então,*

$$C(A, B|E) \stackrel{\text{not}}{\equiv} (A - \{E\}) \cup (B - \{E^c\})$$

é denominado o consenso de A e B .

Lema 2.3.1 *Se $C(A, B|E)$ é o consenso de A e B , então*

$$\{A\} \vee \{B\} \equiv \{A\} \vee \{B\} \vee C(A, B|E).$$

Prova -

$$\begin{aligned} (\cap A) \cup (\cap B) &= (\cap A) \cup (\cap B) \cup \\ &\quad \left(\cap \left(\overbrace{E \cup (A - \{E\})}^A \cup (B - \{E^c\}) \right) \right) \cup \\ &\quad \left(\cap \left((A - \{E\}) \cup \overbrace{\{E^c\} \cup (B - \{E^c\})}^B \right) \right) \\ &= (\cap A) \cup (\cap B) \cup \\ &\quad \left(\overbrace{(E \cup E^c)}^{\Omega} \cap \left(\cap \left((A - \{E\}) \cup (B - \{E^c\}) \right) \right) \right). \end{aligned}$$

□

Lema 2.3.2 *Se $A, B \in \delta$ e $B \supset A$, então $\delta \equiv \delta - \{B\}$.*

□

Lema 2.3.3 *Se δ é uma disjunção normal para T e C é suficiente mínimo para T , então, para todo $H \in C$, existe $D \in \delta$ tal que $H \in D$ ou $H^c \in D$.*

Prova -

- Se $C \in \delta$ é óbvio.

- Se $C \notin \delta$, suponha que existe $H \in C$ tal que, para todo $D \in \delta$, $H, H^c \notin D$. Assim,

$$\emptyset = \left(\bigcap_{D \in \delta} \bigcup \overline{D} \right) \cap (\bigcap C) = \bigcup_{F \in X\delta} \left(\bigcap_{D \in \delta} F_D^c \right) \cap (\bigcap C),$$

onde $X\delta$ é o produto cartesiano dos $D \in \delta$, e $F \in X\delta$ é uma função $F : \delta \mapsto \mathcal{E}$ tal que $F(D) = F_D \in D$ para $D \in \delta$.

Logo, $\left(\bigcap_{D \in \delta} F_D^c \right) \cap (\bigcap C) = \emptyset$, para todo $F \in X\delta$, e como $F_D \neq H$, para todo $D \in \delta$, tem-se $\left(\bigcap_{D \in \delta} F_D^c \right) \cap (\bigcap (C - \{H\})) = \emptyset$, para todo $F \in X\delta$. Portanto, $\emptyset = \left(\bigcap_{D \in \delta} \bigcup \overline{D} \right) \cap (\bigcap (C - \{H\}))$, e conseqüentemente, $C - \{H\}$ é suficiente para T .

□

Lema 2.3.4 *Se δ é uma disjunção normal para T e $C \notin \delta$ é suficiente mínimo para T , existem $A, B \in \delta$ tais que $A \cap \overline{B} = \{E\}$ e $\delta \equiv \delta \vee \{C(A, B|E)\}$.*

Próva - Como C é suficiente mínimo para T e $C \notin \delta$, então existe pelo menos um conjunto G , suficiente para T , satisfazendo:

1. $G \supseteq C$;
2. para todo $A \in \delta$, $G \not\supseteq A$;
3. para todo $H \in G$, existe $D \in \delta$ tal que $H \in D$ ou $H^c \in D$.

Seja K o conjunto com o maior número de elementos que satisfaz 1. a 3.. Nestas condições, existe $E \in \bigcup \delta$ tal que $E, E^c \notin K$, pois suponha o contrário, ou seja, que para todo $E \in \bigcup \delta$, $E \in K$ ou $E^c \in K$. Assim:

- i. existe $D \in \delta$ tal que para todo $E \in D$, $E \in K$, o que violaria 2, pois neste caso $D \subseteq K$;
- ii. caso contrário, para todo $D \in \delta$ existe $E \in D$ tal que $E^c \in K$, o que violaria 1., ou seja, que K é suficiente para T , pois neste caso, $K \cap \overline{D} \neq \emptyset$, para todo $D \in \delta$.

Logo, existe $E \in (\cup \delta)$ tal que $E, E^c \notin K$. Como K é o conjunto com o maior número de elementos que satisfaz 1. a 3., $K \cup \{E\}$ e $K \cup \{E^c\}$ violam um dos itens de 1. a 3.. É fácil ver que só pode ser o item 2. Portanto, existem $A, B \in \delta$ tal que $(K \cup \{E\}) \supseteq A$ e $(K \cup \{E^c\}) \supseteq B$. Por 2., $K \not\supseteq A$ e $K \not\supseteq B$. Como $(K \cup \{E\}) \supseteq A$ e $(K \cup \{E^c\}) \supseteq B$, então $E \in A$ e $E^c \in B$. Além disso, não existem $I \in A$ e $I^c \in B$ ($I, I^c \neq E, E^c$) ou vice-versa. Assim A e B tem um consenso, a menos que sejam identicamente iguais a E e E^c , respectivamente. Mas isto vai contra a hipótese considerada de que $T \in \sigma'(\mathcal{E})$. O resultado segue do Lema 2.3.1.

□

Lema 2.3.5 *Se δ é uma disjunção normal para T e $C \in \delta$ não é suficiente mínimo para T , então:*

- 1) *Existe $A \in \delta$ tal que $A \subset C$; ou*
- 2) *Existem $A, B \in \delta$ tais que $C(A, B|E) \vee \delta \equiv \delta$, $A \cap \bar{B} = \{E\}$.*

Prova - Se C não é suficiente mínimo para T , existe A , suficiente mínimo para T , tal que $A \subset C$. Se $A \in \delta$, vale 1). Se $A \notin \delta$, então 2) segue de 2.3.4.

□

Procedimento do Consenso - Se δ é uma disjunção normal para T , então

- a) Para $A, C \in \delta$, se $C \supset A$, elimine C de δ .
- b) Se C é o consenso de $A, B \in \delta$, inclua C em δ , a menos que $C \supseteq D$, para algum $D \in \delta$.

Teorema 2.3.1 *O Procedimento do Consenso aplicado a uma disjunção normal δ para T gera a base completa para T .*

Prova - Segue imediatamente dos Lemas 2.3.1 a 2.3.5.

□

Utilizando o resultado do Teorema 2.3.1, pode-se obter a base completa para o evento topo T , de uma árvore de eventos \mathcal{A} , através do algoritmo de Quine, executando as seguintes etapas:

Qui.1 caso não se disponha de uma disjunção normal δ para o evento topo T , derive-a através do algoritmo DN ou do algoritmo de Fussell, Henry & Marshall (Seção 2.3.1);

Qui.2 seja $\mathcal{B} = \delta$ a disjunção normal disponível para T :

(2.1) obtenha $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}})$ e construa $\mathcal{C} = \{(E, E^c) : E, E^c \in (\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}})\}$. Observe que o conjunto obtido corresponde ao conjunto dos pares de eventos terminais básicos que permitem gerar um consenso;

(2.2) para $A, B \in \mathcal{B}$, se $A \supset B$, elimine A de \mathcal{B} ;

(2.3) utilizando um par $(E, E^c) \in \mathcal{C}$ de cada vez, forme os possíveis consensos a partir de (E, E^c) , utilizando os conjuntos de \mathcal{B} que contém E ou E^c . Se C é o consenso de $A, B \in \mathcal{B}$, inclua C em \mathcal{B} , a menos que $C \supseteq D$, para algum $D \in \mathcal{B}$.

repita (2.2) e (2.3) na ordem exposta, até que não seja mais possível executá-las. A disjunção normal \mathcal{B} obtida no final deste procedimento, é a base completa para T .

Observação: Não é difícil ver que os conjuntos eliminados de \mathcal{B} por (2.2) não são novamente incluídos em \mathcal{B} por (2.3); isto é, não há possibilidade do algoritmo não terminar.

Exemplo 2.3.6 - A aplicação do algoritmo de Quine à árvore de eventos \mathcal{A} , definida no Exemplo 2.3.5, fornece o seguinte resultado:

Etapa Qui.1

Etapa DN.1 Através do método “top-down”:

$$\begin{aligned} T &= E_{21} \cup E_{22} \\ &= (Q \cap E_{32}) \cup (Q^c \cap R \cap S^c) \\ &= [Q \cap (P \cup S \cup R^c)] \cup [Q^c \cap R \cap S^c]. \end{aligned}$$

Etapa DN.2

$$\begin{aligned} T &= [Q \cap (P \cup S \cup R^c)] \cup [Q^c \cap R \cap S^c] \\ &= (Q \cap P) \cup (Q \cap S) \cup (Q \cap R^c) \cup (Q^c \cap R \cap S^c). \end{aligned}$$

Logo,

$$\mathcal{B} = \delta = \{\{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{Q^c, R, S^c\}\}.$$

Etapa Qui.2 $\mathcal{C} = \{(Q, Q^c), (R, R^c), (S, S^c)\}$.

Utilizando o par (Q, Q^c) , o único consenso possível é $\{P, R, S^c\}$. Logo,

$$\begin{aligned} \mathcal{B} &= \{\{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{Q^c, R, S^c\}\} \\ &\equiv \{\{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{Q^c, R, S^c\}, \{P, R, S^c\}\}. \end{aligned}$$

Utilizando o par (R, R^c) , o único consenso possível é $\{Q, P, S^c\} \subset \{Q, P\} \in \mathcal{B}$. Logo, o consenso não é adicionado à \mathcal{B} .

Utilizando o par (S, S^c) , o único consenso possível é $\{Q, P, R\} \subset \{Q, P\} \in \mathcal{B}$. Logo, o consenso não é adicionado à \mathcal{B} .

Assim,

$$\mathcal{B} = \{\{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{Q^c, R, S^c\}, \{P, R, S^c\}\},$$

é a base completa para T .

Algoritmo de Nelson

Nelson [Nel] propôs um algoritmo para a obtenção da base completa para o evento topo T de uma árvore de eventos \mathcal{A} , a partir de uma conjunção normal ϱ para T . Como $T = \bigcap_{C \in \varrho} \bigcup C$, seu algoritmo se baseia no fato de que para $C \in \varrho$, $\{\{E\} : E \in C\}$ pode ser encarada como a base completa para o evento $\bigcup C = B$, já que todo elemento de $\{\{E\} : E \in C\}$ é suficiente mínimo para B e todo conjunto suficiente mínimo para B está em $\{\{E\} : E \in C\}$ (veja Definição 2.2.7 - Seção 2.2). Isto sugere que os conjuntos suficientes mínimos para T são conjuntos formados por um elemento de cada conjunto $C \in \varrho$.

Defina as disjunções:

$$K = \bigwedge_{C \in \varrho} \{\{E\} : E \in C\}$$

e

$$K^\circ = \{D \in K : D \not\supseteq B, \text{ para todo } B \neq D, B \in K\}.$$

Lema 2.3.6 *Se $A \in K$, então A é suficiente para T .*

Prova - Se $A \in K$, então $A \cap C \neq \emptyset$, para todo $C \in \varrho$. Logo, $\bigcap A \subseteq \bigcup C$, para todo $C \in \varrho$. Portanto, $\bigcap A \subseteq \bigcap_{C \in \varrho} \bigcup C = T$. Consequentemente, A é suficiente para T .

□

Lema 2.3.7 *Se L é suficiente para T , então existe $A \in K$ tal que $\bigcap L \subseteq \bigcap A$.*

Prova - Se L é suficiente para T , $\bigcap L \subseteq \bigcap_{C \in \varrho} \bigcup C$. Logo, $\bigcap L \subseteq \bigcup C$, para todo $C \in \varrho$ e assim, $L \cap C \neq \emptyset$, para todo $C \in \varrho$. Consequentemente, pela definição de K , existe um $A \in K$ tal que $\bigcap L \subseteq \bigcap A$.

□

Lema 2.3.8 *Se L é suficiente mínimo para T , então $L \in K$.*

Prova - Pelo Lema 2.3.7, existe $A \in K$ tal que $\cap L \subseteq \cap A$. Pelo Lema 2.3.6, A é suficiente para T . Como L é suficiente mínimo para T , pela definição de conjunto suficiente mínimo, $L = A$, ou seja, $L \in K$.

□

Teorema 2.3.2 *L é suficiente mínimo para T se, e somente se, $L \in K^\circ$.*

Prova -

(\Rightarrow)

Pelo Lema 2.3.8, $L \in K$. Como L é suficiente mínimo para T , pela definição de K° , segue que $L \in K^\circ$.

(\Leftarrow)

Se existir $L' \subset L$, L' suficiente para T , pelo Lema 2.3.7 existe $A \in K$ tal que $\cap L' \subseteq \cap A$ e conseqüentemente, $L \supset A$ com $A \neq L$. Pela definição de K° , $L \notin K^\circ$, o que é absurdo. Portanto, L é suficiente mínimo para T .

□

Utilizando o resultado do Teorema 2.3.2, pode-se executar o algoritmo de Nelson, através das seguintes etapas:

- Nel.1 caso não se disponha de uma conjunção normal para T , derive-a utilizando o algoritmo CN; caso contrário, vá para a etapa seguinte;
- Nel.2 utilizando a conjunção normal disponível para T , aplique o algoritmo DN à expressão algébrica para T representada através desta conjunção normal. O resultado é uma expressão da forma $T = \bigcup_{C \in \mathcal{B}} \cap C$, onde \mathcal{B} é a base completa de T .

Exemplo 2.3.7 - A aplicação do algoritmo de Nelson à árvore de eventos \mathcal{A} , definida no Exemplo 2.3.5, fornece o seguinte resultado:

Etapa Nel.1

Etapa CN.1 Através do método “top-down”:

$$\begin{aligned} T &= E_{21} \cup E_{22} \\ &= (Q \cap E_{32}) \cup (Q^c \cap R \cap S^c) \\ &= [Q \cap (P \cup S \cup R^c)] \cup [Q^c \cap R \cap S^c]. \end{aligned}$$

Etapa CN.2

$$\begin{aligned} T^c &= [Q^c \cup (P^c \cap S^c \cap R)] \cap [Q \cup R^c \cup S] \\ &= (Q^c \cap Q) \cup (Q^c \cap R^c) \cup (Q^c \cap S) \cup \\ &\quad (P^c \cap S^c \cap R \cap Q) \cup (P^c \cap S^c \cap R \cap R^c) \cup (P^c \cap S^c \cap R \cap S) \\ &= (Q^c \cap R^c) \cup (Q^c \cap S) \cup (P^c \cap S^c \cap R \cap Q). \end{aligned}$$

Etapa CN.3

$$T = (Q \cup R) \cap (Q \cup S^c) \cap (P \cup S \cup R^c \cup Q^c).$$

Etapa Nel.2

Etapa DN.2

$$\begin{aligned} T &= (Q \cup R) \cap (Q \cup S^c) \cap (P \cup S \cup R^c \cup Q^c) \\ &= [(Q \cap Q) \cup (Q \cap S^c) \cup (Q \cap R) \cup (R \cap S^c)] \cap \\ &\quad [P \cup S \cup R^c \cup Q^c] \\ &= [Q \cup (R \cap S^c)] \cap [P \cup S \cup R^c \cup Q^c] \\ &= (Q \cap P) \cup (Q \cap S) \cup (Q \cap R^c) \cup \\ &\quad (Q \cap Q^c) \cup (P \cap R \cap S^c) \cup (R \cap S^c \cap S) \cup \\ &\quad (R \cap S^c \cap R^c) \cup (Q^c \cap R \cap S^c) \\ &= (Q \cap P) \cup (Q \cap S) \cup (Q \cap R^c) \cup (P \cap R \cap S^c) \cup (Q^c \cap R \cap S^c). \end{aligned}$$

Logo, a base completa para T é

$$\mathcal{B} = \{\{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{P, R, S^c\}, \{Q^c, R, S^c\}\}.$$

Observações:

- Hulme & Worrell [HW] observaram que a aplicação do algoritmo DN, aos complementares das expressões para um evento topo, nas formas original e fatorada, fornece disjunções normais idênticas. Com base neste fato, propuseram um algoritmo para a obtenção da base completa para o evento topo T , de uma árvore de eventos \mathcal{A} , acelerando o procedimento do algoritmo de Nelson, realizando, na etapa Ne1.1, a fatoração da expressão disponível para o evento topo de interesse, antes de executar a etapa CN.2, e a fatoração da expressão obtida em CN.2, antes de executar a etapa CN.3.

Para se ter uma idéia do sentido desta observação, seja $D = (A \cap B) \cup (A \cap C)$, onde $A, B, C \in \mathcal{E}$. Aplicando as leis de De Morgan e as leis distributivas à expressão de D^c , tem-se

$$\begin{aligned} D^c &= (A^c \cup B^c) \cap (A^c \cup C^c) \\ &= (A^c \cap A^c) \cup (A^c \cap C^c) \cup (A^c \cap B^c) \cup (B^c \cap C^c), \end{aligned}$$

que se reduz à

$$D^c = A^c \cup (B^c \cap C^c),$$

ao final da aplicação do algoritmo DN à expressão de D^c .

Esta última expressão é obtida de forma mais imediata fatorando-se A na expressão original de D , ou seja, fazendo $D = A \cap (B \cup C)$ e tomando-se o complementar;

- Hulme & Worrell [HW] implementaram em um CDC6600, os algoritmos de Nelson e Nelson com fatoração, no programa SETS ([Wor1], [Wor2], [WS]), comparando a eficiência (em segundos de CPU) dos dois algoritmos, através de vários exemplos. Os resultados obtidos demonstraram a superioridade do algoritmo de Nelson com fatoração com relação ao algoritmo de Nelson.

Algoritmo de Kumamoto & Henley

O algoritmo proposto por Kumamoto & Henley [KH] é um algoritmo que percorre a estrutura lógica da árvore de eventos dual, \mathcal{A}^D , no sentido “top-down”, em busca dos conjuntos suficientes mínimos para o evento topo T de \mathcal{A} . Esta busca se dá através de aplicações sucessivas de operadores sobre disjunções, que são geradas pela própria aplicação dos operadores e/ou por refinamentos convenientes de eventos porta de \mathcal{A}^D .

O grande valor do algoritmo de Kumamoto & Henley está na obtenção de bases completas para eventos topo de árvores de eventos cujas estruturas são muito complexas. Nestas situações, a superioridade do algoritmo de Kumamoto & Henley, quando comparado com os demais algoritmos para o mesmo fim, consiste fundamentalmente de dois aspectos: o primeiro está na propriedade de que seus operadores, aplicados à disjunções equivalentes, geram disjunções equivalentes. Este fato possibilita trabalhar com disjunções não-normais, equivalentes a disjunções normais, constituídas por um número menor de elementos; o segundo é que operando sobre a árvore de eventos dual \mathcal{A}^D , o algoritmo permite que as operações sejam feitas sobre disjunções, ao invés de conjunções, já que são propriedades destas últimas que dão suporte teórico para a validade do algoritmo. A vantagem dessa possibilidade está em que a obtenção de disjunções é geralmente mais simples que a de conjunções.

O desenvolvimento do algoritmo é baseado em algumas propriedades dos operadores que são definidos a seguir.

Definição 2.3.2 Para $Z, U \in \mathcal{P}(\mathcal{E})$ tais que $Z \cap (\bar{Z} \cup U) = \emptyset$, seja $\langle Z, U \rangle$ o operador definido pela transformação

$$\delta \in \mathcal{D} \xrightarrow{\langle Z, U \rangle} \delta \langle Z, U \rangle = \bigvee_{C \in \delta} \{C \langle Z, U \rangle\} \in \mathcal{D},$$

onde $C \langle Z, U \rangle$, para $C \in \mathcal{P}^*(\mathcal{E})$, é definido por:

$$C \langle Z, U \rangle = \begin{cases} C & \text{se } C \cap (Z \cup \bar{Z} \cup U) = \emptyset, \\ \{\emptyset\} & \text{se } C \cap Z \neq \emptyset, \\ C - (\bar{Z} \cup U) & \text{se } C \cap (\bar{Z} \cup U) \neq \emptyset \text{ e } C \not\subseteq (\bar{Z} \cup U), \\ \{\Omega\} & \text{se } C \subseteq (\bar{Z} \cup U). \end{cases}$$

O resultado da Proposição 2.3.1 segue imediatamente das definições 2.2.5 e 2.3.2.

Proposição 2.3.1 Se $\delta, \xi \in \mathcal{D}$, então $(\delta \vee \xi)\langle Z, U \rangle = \delta \langle Z, U \rangle \vee \xi \langle Z, U \rangle$.

□

Proposição 2.3.2 Se $\delta \in \mathcal{D}$, $Z_2 \cap (\overline{Z_1} \cup U_1 \cup \overline{Z_2} \cup U_2) = \emptyset$ e $U_2 \cap Z_1 = \emptyset$, então

$$(\delta \langle Z_1, U_1 \rangle) \langle Z_2, U_2 \rangle \equiv \delta \langle Z_1 \cup Z_2, U_1 \cup U_2 \rangle.$$

□

Observação:

- Observe que as condições impostas para a validade da Proposição 2.3.2 são as condições necessárias para que o operador $\langle Z_1 \cup Z_2, U_1 \cup U_2 \rangle$ esteja bem definido, dado que o operador $\langle Z_1, U_1 \rangle$ está bem definido.

Proposição 2.3.3 Se $\delta, \xi \in \mathcal{D}$ e $\delta \equiv \xi$, então $\delta \langle Z, U \rangle \equiv \xi \langle Z, U \rangle$.

Prova - Tendo em vista o resultado da Proposição 2.3.2, basta mostrar que o resultado vale para $\langle Z, U \rangle$ da forma $\langle \{E\}, \emptyset \rangle$ ou $\langle \emptyset, \{E\} \rangle$, com $E \in \mathcal{E}$.

Suponha que $(\bigcup_{C \in \delta} \cap C \langle \{E\}, \emptyset \rangle) \cap (\bigcap_{D \in \xi} \cup \overline{D} \langle \{E\}, \emptyset \rangle) \neq \emptyset$. Como

$$(\bigcup_{C \in \delta} \cap C \langle \{E\}, \emptyset \rangle) \cap (\bigcap_{D \in \xi} \cup \overline{D} \langle \{E\}, \emptyset \rangle) = \bigcup_{A \in \varphi} \cap A,$$

existe $A \neq \emptyset \in \varphi$, tal que $\cap A \neq \emptyset$ e $E, E^c \notin A$. Logo, $\cap I = \cap (A \cup E^c) \neq \emptyset$. Por outro lado,

$$\cap I = (\bigcup_{A \in \varphi} \cap A) \cap (\cap I)$$

$$\begin{aligned}
 &\subseteq \left(\bigcup_{C \in \delta} \bigcap C(\{E\}, \emptyset) \right) \cap (\cap I) \\
 &= \bigcup_{C \in \delta} \bigcap (C(\{E\}, \emptyset) \cup I) \\
 &= \bigcup_{C \in \delta} \bigcap (C \cup I) \\
 &= \left(\bigcup_{C \in \delta} \bigcap C \right) \cap (\cap I) \\
 &= \left(\bigcup_{D \in \xi} \bigcap D \right) \cap (\cap I) \\
 &= \bigcup_{D \in \xi} \bigcap (D \cup I) \\
 &= \bigcup_{D \in \xi} \bigcap (D(\{E\}, \emptyset) \cup I) \\
 &= \left(\bigcup_{D \in \xi} \bigcap D(\{E\}, \emptyset) \right) \cap (\cap I) \\
 &= \emptyset,
 \end{aligned}$$

o que leva a contradição de que $\cap I = \emptyset$ e $\cap I \neq \emptyset$. Portanto, $\delta \langle \{E\}, \emptyset \rangle \equiv \xi \langle \{E\}, \emptyset \rangle$.

□

Definição 2.3.3 Para $Z, U \in \mathcal{P}(\mathcal{E})$ tais que $Z \cap (\overline{Z} \cup U) = \emptyset$, seja $\prec Z, U \succ$ o operador definido pela transformação

$$\delta \in \mathcal{D}^\circ \xrightarrow{\prec Z, U \succ} \delta \prec Z, U \succ = \bigvee_{C \in \delta} \mathcal{B}_{nC} \langle Z, U \rangle \in \mathcal{D}^\circ.$$

Proposição 2.3.4 Se $\delta \in \mathcal{D}$, então $\delta \prec Z, U \succ = \delta \langle Z, U \rangle$.

Prova - Como $\{C\} = \mathcal{B}_{nC}$, para todo $C \in \delta$, $\delta = \bigvee_{C \in \delta} \mathcal{B}_{nC}$. Logo, pela Proposição 2.3.1, $\delta \langle Z, U \rangle = \bigvee_{C \in \delta} \mathcal{B}_{nC} \langle Z, U \rangle = \delta \prec Z, U \succ$.

□

Proposição 2.3.5 Se $\delta \in \mathcal{D}^\circ$, $\xi \in \mathcal{D}$ e $\delta \equiv \xi$, então $\delta \prec Z, U \succ \equiv \xi \langle Z, U \rangle$.

Prova - Como $\{C\} \equiv \mathcal{B}_{nC}$, para todo $C \in \delta$, pela Proposição 2.2.1, $\delta \equiv \bigvee_{C \in \delta} \mathcal{B}_{nC}$. Por outro lado, $\delta \equiv \xi$, portanto, $\xi \equiv \bigvee_{C \in \delta} \mathcal{B}_{nC}$. Logo, pelas Proposições 2.3.1 e 2.3.3,

$$\xi \langle Z, U \rangle \equiv \bigvee_{C \in \delta} \mathcal{B}_{nC} \langle Z, U \rangle = \delta \prec Z, U \succ .$$

□

Proposição 2.3.6 Se $\delta, \xi \in \mathcal{D}^\circ$ e $\delta \equiv \xi$, então $\delta \prec Z, U \succ \equiv \xi \prec Z, U \succ$.

Prova - Como $\{C\} \equiv \mathcal{B}_{nC}$, para todo $C \in \delta$, $\{D\} \equiv \mathcal{B}_{nD}$, para todo $D \in \xi$ e $\delta \equiv \xi$, pela Proposição 2.2.1,

$$\bigvee_{C \in \delta} \mathcal{B}_{nC} \equiv \delta \equiv \xi \equiv \bigvee_{D \in \xi} \mathcal{B}_{nD}.$$

Logo, pelas Proposições 2.3.1 e 2.3.3,

$$\bigvee_{C \in \delta} \mathcal{B}_{nC} \langle Z, U \rangle \equiv \bigvee_{D \in \xi} \mathcal{B}_{nD} \langle Z, U \rangle.$$

Consequentemente,

$$\delta \prec Z, U \succ \equiv \xi \prec Z, U \succ .$$

□

Proposição 2.3.7 Se $C, D \in \mathbf{F}^*(\mathcal{E})$, então $\{C\} \wedge \{D\} \equiv \{C\} \wedge \{D \langle \bar{C}, \emptyset \rangle\}$.

Prova -

1. Se $D \cap \bar{C} = \emptyset$ e

1.1. $D \cap C = \emptyset$, então $D \langle \bar{C}, \emptyset \rangle = D$. Logo,

$$\{C\} \wedge \{D\} = \{C\} \wedge \{D \langle \bar{C}, \emptyset \rangle\};$$

1.2. $D \cap C \neq \emptyset$ e $D \not\subseteq C$, então $D\langle \bar{C}, \emptyset \rangle = D - C$. Logo,

$$\begin{aligned} \{C\} \wedge \{D\} &= \{C \cup D\} \\ &= \{C \cup (D - C)\} \\ &= \{C \cup D\langle \bar{C}, \emptyset \rangle\} \\ &= \{C\} \wedge \{D\langle \bar{C}, \emptyset \rangle\}; \end{aligned}$$

1.3. $D \cap C \neq \emptyset$ e $D \subseteq C$, então $D\langle \bar{C}, \emptyset \rangle = \{\Omega\}$. Logo,

$$\begin{aligned} \{C\} \wedge \{D\} &= \{C \cup D\} \\ &= \{C\} \\ &\equiv \{C \cup \{\Omega\}\} \\ &= \{C\} \wedge \{D\langle \bar{C}, \emptyset \rangle\}. \end{aligned}$$

2. Se $D \cap \bar{C} \neq \emptyset$, então $D\langle \bar{C}, \emptyset \rangle = \{\emptyset\}$. Logo,

$$\begin{aligned} \{C\} \wedge \{D\} &= \{\{\emptyset\}\} \\ &\equiv \{C \cup \{\emptyset\}\} \\ &= \{C\} \wedge \{D\langle \bar{C}, \emptyset \rangle\}. \end{aligned}$$

□

Proposição 2.3.8 Se $A, B \in \mathcal{P}^*(\mathcal{E}_0)$ e $B \cap \bar{A} = \emptyset$, então

$$(\{A\} \wedge \{B\})\langle Z, U \rangle \equiv \{A\langle Z, U \rangle\} \wedge \{B\langle Z, U \rangle\}.$$

Prova -

Basta observar que

$$(\{A\} \wedge \{B\})\langle Z, U \rangle = \{(A \cup B)\langle Z, U \rangle\},$$

que

$$\{A\langle Z, U \rangle\} \wedge \{B\langle Z, U \rangle\} = \{A\langle Z, U \rangle \cup B\langle Z, U \rangle\},$$

e que

$$\cap\left((A \cup B)\langle Z, U \rangle \right) = \cap\left(A\langle Z, U \rangle \cup B\langle Z, U \rangle \right).$$

□

Proposição 2.3.9 *Se $A \in \mathbf{P}^*(\mathcal{E})$ e $B \in \mathbf{P}^*(\sigma'(\mathcal{E}) - \mathcal{E})$, então*

$$(\{A\} \wedge \{B\}) \prec Z, U \succ \equiv \{A\langle Z, U \rangle\} \wedge \left((\{B\} \prec \bar{A}, \emptyset \succ) \prec Z, U \succ \right).$$

Prova - Como $\{B\} \equiv \mathcal{B}_{\cap B}$, pela Proposição 2.2.2, $\{A\} \wedge \{B\} \equiv \{A\} \wedge \mathcal{B}_{\cap B}$. Logo,

$$\begin{aligned} (\{A\} \wedge \{B\}) \prec Z, U \succ &\equiv \left(\{A\} \wedge \mathcal{B}_{\cap B} \right) \langle Z, U \rangle \quad (\text{Prop. 2.3.5}) \\ &= \bigvee_{D \in \mathcal{B}_{\cap B}} \left(\{A\} \wedge \{D\} \right) \langle Z, U \rangle \quad (\text{Prop. 2.3.1}) \\ &\equiv \bigvee_{D \in \mathcal{B}_{\cap B}} \left(\{A\} \wedge \{D\langle \bar{A}, \emptyset \rangle\} \right) \langle Z, U \rangle \quad (\text{Prop. 2.3.7}) \\ &\equiv \bigvee_{D \in \mathcal{B}_{\cap B}} \left(\{A\langle Z, U \rangle\} \wedge \left(\{D\langle \bar{A}, \emptyset \rangle\} \right) \langle Z, U \rangle \right) \quad (\text{Prop. 2.3.8}) \\ &= \{A\langle Z, U \rangle\} \wedge \left(\bigvee_{D \in \mathcal{B}_{\cap B}} \left(\{D\langle \bar{A}, \emptyset \rangle\} \right) \langle Z, U \rangle \right) \\ &= \{A\langle Z, U \rangle\} \wedge \left(\left(\bigvee_{D \in \mathcal{B}_{\cap B}} \{D\langle \bar{A}, \emptyset \rangle\} \right) \langle Z, U \rangle \right) \quad (\text{Prop. 2.3.1}) \\ &= \{A\langle Z, U \rangle\} \wedge \left(\left(\mathcal{B}_{\cap B} \langle \bar{A}, \emptyset \rangle \right) \langle Z, U \rangle \right) \\ &= \{A\langle Z, U \rangle\} \wedge \left(\left(\mathcal{B}_{\cap B} \langle \bar{A}, \emptyset \rangle \right) \prec Z, U \succ \right) \quad (\text{Prop. 2.3.4}) \\ &\equiv \{A\langle Z, U \rangle\} \wedge \left((\{B\} \prec \bar{A}, \emptyset \succ) \prec Z, U \succ \right) \quad (\text{Prop. 2.3.5}). \end{aligned}$$

□

Para facilitar a manipulação com disjunções na execução do algoritmo de Kumamoto & Henley, serão utilizadas formas especiais de representação. Observe que como para toda disjunção $\delta \in \mathcal{D}^\circ$ se tem

$$\delta = \bigvee_{C \in \delta} (\{C \cap \mathcal{E}\} \wedge \{C - \mathcal{E}\}),$$

a Proposição 2.3.9 garante a existência de uma disjunção equivalente à $\delta \prec Z, U \succ$ da forma

$$\bigvee_{i=1}^{n_{\delta \prec Z, U \succ}} \{C_i\} \wedge ((\{D_i\} \prec X_i, \emptyset \succ) \prec Z, U \succ), \quad (2.3.2)$$

onde, para $1 \leq i \leq n_{\delta \prec Z, U \succ}$:

- $C_i \in \mathcal{P}^*(\mathcal{E}_o)$ ou $\{C_i\} = \emptyset$,
- $D_i \in \mathcal{P}^*(\sigma'(\mathcal{E}) - \mathcal{E})$ ou $\{D_i\} = \emptyset$,
- $\bar{C}_i \subseteq X_i$,
- $C_i \cap (Z \cup \bar{Z} \cup U) = \emptyset$.

Observação :

- Para simplificar a notação, $\{C_i\} = \emptyset$ ($\{D_i\} = \emptyset$) será denotado por $C_i = \emptyset$ ($D_i = \emptyset$).

Para representar simbolicamente a equivalência entre $\delta \prec Z, U \succ$ e (2.3.2), será utilizada a forma

$$\{(C_i, D_i, X_i) : 1 \leq i \leq n_{\delta \prec Z, U \succ}\} \delta \prec Z, U \succ. \quad (2.3.3)$$

A forma (2.3.3) será denominada:

- **não-ativa** se $C_i \cap \mathcal{E} = \emptyset$, para $i = 1, \dots, n_{\delta \prec Z, U \succ}$ e $D_i \neq \emptyset$, para algum $i = 1, \dots, n_{\delta \prec Z, U \succ}$;

- **ativa** se $C_i \cap \mathcal{E} \neq \emptyset$, para algum $i = 1, \dots, n_{\delta \prec Z, U \succ}$;
- **terminal** se $(C_i, D_i, X_i) = (\{\emptyset\}, D_i, X_i)$, para $i = 1, \dots, n_{\delta \prec Z, U \succ}$ ou se $(C_i, D_i, X_i) = (\{\Omega\}, \emptyset, \emptyset)$, para algum $i = 1, \dots, n_{\delta \prec Z, U \succ}$.

Quando (2.3.3) é terminal e $(C_i, D_i, X_i) = (\{\emptyset\}, D_i, X_i)$, para $i = 1, \dots, n_{\delta \prec Z, U \succ}$, $\delta \prec Z, U \succ \in [\emptyset]$, onde $[\emptyset]$ representa a classe de equivalência das disjunções para o evento \emptyset . Quando (2.3.3) é terminal e $(C_i, D_i, X_i) = (\{\Omega\}, \emptyset, \emptyset)$, para algum $i = 1, \dots, n_{\delta \prec Z, U \succ}$, $\delta \prec Z, U \succ \in [\Omega]$, onde $[\Omega]$ representa a classe de equivalência das disjunções para o evento Ω .

Dada uma árvore \mathcal{A} qualquer, pode-se, introduzindo se necessário, eventos porta adicionais, reescrever \mathcal{A} de maneira que em sua estrutura só figurem eventos porta “NAND”, “NOR”, “AND”, “OR” e “NOT”. Para a execução do algoritmo de Kumamoto & Henley, supõe-se que a árvore \mathcal{A} tem esta propriedade.

Com base nas Proposições 2.3.1 a 2.3.9 e nas definições 2.3.2 e 2.3.3, o algoritmo de Kumamoto & Henley para a obtenção da base completa para o evento topo T , de uma árvore de eventos \mathcal{A} , consiste das seguintes etapas:

KH.1 obtenha as expressões locais para cada evento porta da árvore de eventos \mathcal{A} , reescrevendo cada expressão local especial como uma combinação de expressões locais fundamentais (veja Tabelas 1.2.1, 1.3.1 e 1.3.2 - Capítulo 1), logo após a substituição de cada evento porta por sua expressão local. Complemente cada uma das expressões locais obtidas para os eventos porta de \mathcal{A} , obtendo assim as expressões locais fundamentais para os eventos porta de \mathcal{A}^D .

KH.2 faça $Z = U = C_\emptyset = \emptyset$ e $\mathcal{D}_{KH} = \left\{ \{(\emptyset, \{T^e\}, \emptyset)\} \delta \prec Z, U \succ \right\}$;

KH.3 para cada forma não-ativa

$$\{(C_i, D_i, X_i) : 1 \leq i \leq n_{\delta \prec Z, U \succ}\} \delta \prec Z, U \succ \in \mathcal{D}_{KH}, \quad (2.3.4)$$

obtenha uma forma

$$\{(A_i, B_i, W_i) : 1 \leq i \leq m_{\delta \prec Z, U \succ}\} \delta \prec Z, U \succ, \quad (2.3.5)$$

utilizando a equivalência

$${}^n\delta \langle Z, U \rangle \bigvee_{i=1} {}C_i \wedge \left(\{D_i\} \langle X_i, \emptyset \rangle \langle Z, U \rangle \right) \equiv {}^m\delta \langle Z, U \rangle \{A_i\} \wedge \left(\{B_i\} \langle W_i, \emptyset \rangle \langle Z, U \rangle \right), \quad (2.3.6)$$

onde a expressão do lado direito de (2.3.6) é obtida através do seguinte procedimento:

Procedimento Ativa

Seja $F \in \bigcup_{i=1} {}^n\delta \langle Z, U \rangle D_i$, o evento porta que ocupa o nível mais próximo do evento topo T^c , e seja \bar{H} , o conjunto de entradas de F .

Se $F \notin D_i$, então:

$$A_i = C_i,$$

$$B_i = D_i,$$

$$W_i = X_i;$$

Se $F \in D_i$, seja $G_i \subseteq D_i$ tal que, para todo $I \in G_i$, a expressão local de I é da forma $I = \cap J$. Logo, $\{G_i\} \equiv \{L_i\}$, onde $L_i = \{Q : Q \text{ é entrada de } I, \text{ para } I \in G_i\}$.

caso 1. Se $F = \cap H$, então $F \in G_i$. Logo,

$$\{D_i\} \equiv \{L_i \cap \mathcal{E}\} \wedge \{(D_i - G_i) \cup (L_i - \mathcal{E})\}.$$

Assim, na expressão do lado direito de (2.3.6):

- se $\{L_i \cap \mathcal{E}\} = \emptyset$, então:

$$A_i = \emptyset,$$

$$B_i = L_i \cup (D_i - G_i),$$

$$W_i = X_i;$$

- se $\{L_i \cap \mathcal{E}\} \neq \emptyset$, $L_i \cap \bar{L}_i = \emptyset$ e $(L_i \cap \mathcal{E}) \langle X_i, \emptyset \rangle \neq \{\emptyset\}$, então:

$$A_i = \left((L_i \cap \mathcal{E}) \langle X_i, \emptyset \rangle \right) \langle Z, U \rangle,$$

$$B_i = (L_i - \mathcal{E}) \cup (D_i - G_i),$$

$$W_i = \left[\left(\overline{(L_i \cap \mathcal{E}) \langle X_i, \emptyset \rangle} \right) \cap \mathcal{E} \right] \cup X_i, \text{ se } B_i \neq \emptyset; \text{ caso contrário, } W_i = \emptyset.$$

- caso contrário:

$$A_i = \{\emptyset\},$$

$$B_i = \emptyset,$$

$$W_i = \emptyset.$$

caso 2. Se $F = \cup H$, então $F \notin G_i$. Logo,

$$\{D_i\} \equiv \bigvee_{j=1}^k \left(\left((\{H_j\} \cup L_i) \cap \mathcal{E} \right) \wedge \left((D_i - (\{F\} \cup G_i)) \cup ((\{H_j\} \cup L_i) - \mathcal{E}) \right) \right).$$

Assim, na expressão do lado direito de (2.3.6), para $j = 1, \dots, k$:

- se $\{(\{H_j\} \cup L_i) \cap \mathcal{E}\} = \emptyset$, então:

$$A_{ij} = \emptyset,$$

$$B_{ij} = (\{H_j\} \cup L_i) \cup (D_i - (\{F\} \cup G_i)),$$

$$W_{ij} = X_i;$$

- Se $\{(\{H_j\} \cup L_i) \cap \mathcal{E}\} \neq \emptyset$, $(\{H_j\} \cup L_i) \cap \overline{(\{H_j\} \cup L_i)} = \emptyset$ e $(\{H_j\} \cup L_i) \langle X_i, \emptyset \rangle \neq \{\emptyset\}$, então:

$$A_{ij} = \left((\{(\{H_j\} \cup L_i) \cap \mathcal{E}\} \langle X_i, \emptyset \rangle) \langle Z, U \rangle \right),$$

$$B_{ij} = ((\{H_j\} \cup L_i) - \mathcal{E}) \cup (D_i - (\{F\} \cup G_i)),$$

$$W_{ij} = \left[\overline{(\{(\{H_j\} \cup L_i) \cap \mathcal{E}\} \langle X_i, \emptyset \rangle)} \cap \mathcal{E} \right] \cup X_i, \text{ se } B_{ij} \neq \emptyset; \text{ caso contrário, } W_{ij} = \emptyset.$$

- caso contrário:

$$A_i = \{\emptyset\},$$

$$B_i = \emptyset,$$

$$W_i = \emptyset,$$

com $L_i = \emptyset$, se $G_i = \emptyset$;

Simplifique a forma obtida, utilizando as seguintes identidades:

$$i) \{\{\emptyset\}\} \wedge ((\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \succ) \equiv \{\{\emptyset\}\}$$

$$ii) \left(\bigvee_{C \in \xi} \{C\} \right) \vee \{\{\emptyset\}\} \equiv \bigvee_{C \in \xi} \{C\}$$

iii) $\{\{\Omega\}\} \wedge \left((\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \succ \right) \equiv (\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \succ$, se $B_i \neq \emptyset$

iv) $\{G\} \vee \left[\{G\} \wedge \left((\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \succ \right) \right] \equiv \{G\}$, onde $G \in \mathcal{P}^*(\mathcal{E})$.

Usando i) e ii), exclua da forma obtida todo trio $(\{\emptyset\}, B_i, W_i)$, se $m_{\delta \prec Z, U \succ} > 1$; usando iii), substitua todo trio $(\{\Omega\}, B_i, W_i)$ por (\emptyset, B_i, W_i) , se $B_i \neq \emptyset$; e usando iv), exclua todo trio (G, B_i, W_i) , sempre que um trio $(G, \emptyset, \emptyset)$ estiver contido na forma obtida.

Se a forma obtida, após as simplificações i) a iv), é terminal, e $\delta \prec Z, U \succ \in [\Omega]$, delete (2.3.4) de \mathcal{D}_{KH} e abandone (2.3.5); se a forma obtida é terminal e $\delta \prec Z, U \succ \in [\emptyset]$, inclua \bar{Z} em C_\emptyset , delete (2.3.4) de \mathcal{D}_{KH} e abandone (2.3.5); se na forma obtida $A_i \cap \mathcal{E} \neq \emptyset$ para algum i , substitua em \mathcal{D}_{KH} , (2.3.4) por (2.3.5); caso contrário, repita o Procedimento Ativa com a forma obtida.

Quando não existirem mais formas não-ativas em \mathcal{D}_{KH} vá para a etapa KH.4;

KH.4 para cada forma ativa $\{(A_i, B_i, W_i) : 1 \leq i \leq m_{\delta \prec Z, U \succ}\} \delta \prec Z, U \succ \in \mathcal{D}_{KH}$, escolha um evento terminal E em algum $A_i \cap \mathcal{E} \neq \emptyset$ e gere duas novas formas

$$\{(A_i \langle Z \cup \{E\}, U \rangle, B_i, W_i) : 1 \leq i \leq m_{\delta \prec Z \cup \{E\}, U \succ}\} \delta \prec Z \cup \{E\}, U \succ \quad (2.3.7)$$

e

$$\{(A_i \langle Z, U \cup \{E\} \rangle, B_i, W_i) : 1 \leq i \leq m_{\delta \prec Z, U \cup \{E\} \succ}\} \delta \prec Z, U \cup \{E\} \succ, \quad (2.3.8)$$

a partir das operações

$$\bigvee_{i=1}^{m_{\delta \prec Z, U \succ}} \left(\{A_i\} \wedge \left((\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \succ \right) \right) \prec \{E\}, \emptyset \succ \quad (2.3.9)$$

e

$$\bigvee_{i=1}^{m_{\delta \prec Z, U \succ}} \left(\{A_i\} \wedge \left((\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \succ \right) \right) \prec \emptyset, \{E\} \succ. \quad (2.3.10)$$

Em (2.3.9)

$$\left(\{A_i\} \wedge \left((\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \succ \right) \right) \prec \{E\}, \emptyset \succ \equiv \{A_i \langle \{E\}, \emptyset \rangle\} \wedge \left((\{B_i\} \prec W_i, \emptyset \succ) \prec Z \cup \{E\}, U \succ \right),$$

e em (2.3.10)

$$\left(\{A_i\} \wedge \left((\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \succ \right) \right) \prec \emptyset, \{E\} \succ \equiv \{A_i \langle \emptyset, \{E\} \rangle\} \wedge \left((\{B_i\} \prec W_i, \emptyset \succ) \prec Z, U \cup \{E\} \succ \right).$$

Simplifique as formas obtidas, utilizando as seguintes identidades:

(4.1)

$$\{\{\emptyset\}\} \wedge ((\{B_i\} \prec W_i, \emptyset \succ) \prec Y, V \succ) \equiv \{\{\emptyset\}\}$$

e

$$\left(\bigvee_{C \in \xi} \{C\} \right) \vee \{\{\emptyset\}\} \equiv \bigvee_{C \in \xi} \{C\},$$

para excluir da forma obtida todo trio $(\{\emptyset\}, B_i, W_i)$, se $m_{\delta \prec Z \cup \{E\}, U \succ} > 1$;

(4.2)

$$\{\{\Omega\}\} \wedge ((\{B_i\} \prec W_i, \emptyset \succ) \prec Y, V \succ) \equiv (\{B_i\} \prec W_i, \emptyset \succ) \prec Y, V \succ,$$

se $B_i \neq \emptyset$. Nestes casos, substitua todo trio $(\{\Omega\}, B_i, W_i)$ por (\emptyset, B_i, W_i) ;

(4.3)

$$\{G\} \vee [\{G\} \wedge ((\{B_i\} \prec W_i, \emptyset \succ) \prec Y, V \succ)] \equiv \{G\},$$

onde $G \in P^*(\mathcal{E})$. Nestes casos, exclua todo trio (G, B_i, W_i) , sempre que um trio $(G, \emptyset, \emptyset)$ estiver contido na forma considerada,

onde $Y = Z \cup \{E\}$ ($Y = Z$) e $V = U$ ($V = U \cup \{E\}$), se a forma (2.3.7) ((2.3.8)) estiver sendo considerada.

Delete $\{(A_i, B_i, W_i) : 1 \leq i \leq m_{\delta \prec Z, U \succ}\} \delta_{\prec Z, U \succ}$ de \mathcal{D}_{KH} .

KH.5 para cada uma das formas, (2.3.7) e (2.3.8), obtidas em KH.4, proceda da seguinte maneira:

(5.1) se a forma (2.3.7) ((2.3.8)) é terminal e $\delta_{\prec Z \cup \{E\}, U \succ}$ ($\delta_{\prec Z, U \cup \{E\} \succ}$) pertence a $[\Omega]$, abandone (2.3.7) ((2.3.8));

(5.2) se a forma (2.3.7) é terminal e $\delta_{\prec Z \cup \{E\}, U \succ}$ pertence a $[\emptyset]$, inclua $\overline{Z \cup \{E\}}$ em C_\emptyset e abandone (2.3.7);

(5.3) se a forma (2.3.7) ((2.3.8)) não é terminal, inclua (2.3.7) ((2.3.8)) em \mathcal{D}_{KH} .

Se (5.3) foi realizado e uma forma ativa (2.3.7) ((2.3.8)) foi incluída em \mathcal{D}_{KH} , volte para KH.4, chamando $A_i \langle Z \cup \{E\}, U \rangle$ ($A_i \langle Z, U \cup \{E\} \rangle$) de A_i e $Z \cup \{E\}$ de Z ($U \cup \{E\}$ de U); Se (5.3) foi realizado e somente formas não-ativas foram incluídas em \mathcal{D}_{KH} , volte para KH.3, chamando m de n , $A_i \langle Z \cup \{E\}, U \rangle$ ($A_i \langle Z, U \cup \{E\} \rangle$) de C_i , B_i de D_i , W_i de X_i e $Z \cup \{E\}$ de Z ($U \cup \{E\}$ de U), se (2.3.7) ((2.3.8)) foi incluído; caso contrário, vá para a etapa KH.6;

KH.6 para $A, B \in C_\emptyset$, elimine A se $A \supset B$. \mathcal{B} , o conjunto resultante, é a base completa para T .

Na etapa KH.5, para cada uma das formas, (2.3.7) e (2.3.8), obtidas em KH.5, está se verificando as informações contidas nas operações

$$(\delta \prec Z, U \succ) \prec \{E\}, \emptyset \succ \quad \text{e} \quad (\delta \prec Z, U \succ) \prec \emptyset, \{E\} \succ,$$

com relação à base completa para T . A natureza das informações que são possíveis de obter ficarão mais claras quando for discutido o Teorema 2.3.3, onde prova-se que todos os conjuntos suficientes mínimos para T são obtidos através deste algoritmo.

Teorema 2.3.3 *Seja $\delta \in \mathcal{D}^\circ$ uma disjunção para T^c e \mathcal{Z} , o conjunto dos pares (Z, U) correspondentes às formas terminais geradas pelo algoritmo de Kumamoto & Henley, isto é, tais que $\delta \prec Z, U \succ \in [\emptyset]$ ou $\delta \prec Z, U \succ \in [\Omega]$. Defina*

$$W_\emptyset = \{(Z, U) \in \mathcal{Z} : \delta \prec Z, U \succ \in [\emptyset]\},$$

$$W_\Omega = \{(Z, U) \in \mathcal{Z} : \delta \prec Z, U \succ \in [\Omega]\},$$

$$C_\emptyset = \{\bar{Z} : (Z, U) \in W_\emptyset\} \text{ e}$$

$$C_\Omega = \{\bar{Z} : (Z, U) \in W_\Omega\}.$$

1. *Todo elemento de C_\emptyset é suficiente para T ;*
2. *Todo conjunto suficiente mínimo para T pertence a C_\emptyset .*

Prova - Seja $\xi \in \mathcal{D}$ tal que $\delta \equiv \xi$.

1.

Se $\bar{Z} \in C_\emptyset$, então $\delta \prec Z, U \succ \equiv \xi \langle Z, U \rangle \in [\emptyset]$. Logo, para todo $B \in \xi$, $B \cap Z \neq \emptyset$ (veja Definição 2.3.2) e portanto, $\bar{B} \cap \bar{Z} \neq \emptyset$. Assim, $\cap \bar{Z} \subseteq \cup \bar{B}$, para todo $B \in \xi$ e consequentemente, $\cap \bar{Z} \subseteq \bigcap_{B \in \xi} \cup \bar{B} = T$.

Como \bar{Z} satisfaz $Z \cap \bar{Z} \neq \emptyset$ e $\bar{Z} \subseteq T$, \bar{Z} é suficiente para T .

2.

Para cada par (Z, U) tal que $Z \cap (\bar{Z} \cup U) = \emptyset$, seja

$$S[Z, U] = \{A \in \mathbf{P}^*(\mathcal{E}) : Z \subseteq A, U \cap A = \emptyset\}.$$

Na execução do algoritmo de Kumamoto & Henley, cada par $(Z, U) \in \mathcal{Z}$ é construído por etapas até eventualmente pertencer a W_\emptyset ou a W_Ω . Na primeira etapa do processo, um evento $E \in \mathcal{E}$ é escolhido e os pares $(\{E\}, \emptyset)$ e $(\emptyset, \{E\})$ são gerados. Utilizando este evento escolhido é possível dividir $\mathbf{P}^*(\mathcal{E})$ em duas classes disjuntas:

$$S[\{E\}, \emptyset] = \{A \in \mathbf{P}^*(\mathcal{E}) : \{E\} \subseteq A\} \text{ e } S[\emptyset, \{E\}] = \{A \in \mathbf{P}^*(\mathcal{E}) : \{E\} \cap A = \emptyset\}.$$

Logo,

$$\mathbf{P}^*(\mathcal{E}) = S[\{E\}, \emptyset] + S[\emptyset, \{E\}].$$

Na etapa seguinte, escolhe-se um evento $F \in \mathcal{E}$, tal que $F \neq E$, E^c , gerando os pares $(\{E, F\}, \emptyset)$ e $(\{E\}, \{F\})$. Utilizando este novo evento, é possível dividir $S[\{E\}, \emptyset]$ em duas classes disjuntas. Assim,

$$S[\{E\}, \emptyset] = S[\{E, F\}, \emptyset] + S[\{E\}, \{F\}].$$

De forma análoga, através da escolha de um evento $G \in \mathcal{E}$, tal que $G \neq E$, os pares $(\{G\}, \{E\})$ e $(\emptyset, \{E, G\})$ são gerados. E assim, $S[\emptyset, \{E\}]$ pode ser dividida em duas classes disjuntas:

$$S[\emptyset, \{E\}] = S[\{G\}, \{E\}] + S[\emptyset, \{E, G\}].$$

Portanto, pode-se reescrever $\mathbf{P}^*(\mathcal{E})$ como

$$\mathbf{P}^*(\mathcal{E}) = S[\{E, F\}, \emptyset] + S[\{E\}, \{F\}] + S[\{G\}, \{E\}] + S[\emptyset, \{E, G\}].$$

Este processo continua até que cada par (Z, U) pertença a $(W_\emptyset \cup W_\Omega)$. Dessa forma, $\mathbf{P}^*(\mathcal{E})$ pode ser escrito como

$$\mathbf{P}^*(\mathcal{E}) = \left(\sum_{(Z, U) \in W_\emptyset} S[Z, U] \right) + \left(\sum_{(Z, U) \in W_\Omega} S[Z, U] \right).$$

Pela definição de $S[Z, U]$, se $E \in S[Z, U]$ tem-se: $E \cap U = \emptyset$ e $E \cap \bar{E} = \emptyset$. Como $Z \subseteq E$, então $E \cap \bar{Z} = \emptyset$. Logo, $E \cap (\bar{Z} \cup U) = \emptyset$.

Suponha que $E \in S[Z, U]$ e $(Z, U) \in W_\Omega$. Logo, $\delta \prec Z, U \succ \equiv \xi \langle Z, U \rangle \in [\Omega]$. Assim, existe um $B \in \xi$ tal que $B \subseteq (\overline{Z} \cup U)$ (veja Definição 2.3.2). Como $B \subseteq (\overline{Z} \cup U)$ e $E \cap (\overline{Z} \cup U) = \emptyset$, então $E \cap B = \emptyset$, e portanto $(\cap \overline{E}) \cap (\cap B) \neq \emptyset$. Logo, $\cap \overline{E} \not\subseteq \cup \overline{B}$. Então, $\cap \overline{E} \not\subseteq \bigcap_{B \in \xi} \cup \overline{B} = T$. Dessa forma, \overline{E} não é suficiente para T e portanto, $\overline{Z} \subseteq \overline{E}$ não é suficiente para T . Conseqüentemente, todo conjunto suficiente mínimo para T pertence a C_\emptyset .

□

Através do Teorema 2.3.3, pode-se notar que na etapa KH.4, a disjunção gerada em (2.3.7) vai formar a classe dos conjuntos suficientes mínimos para T que contém o complementar do evento terminal escolhido e a gerada em (2.3.8), vai formar a classe dos conjuntos suficientes mínimos para T que não contém o complementar do evento terminal escolhido.

Exemplo 2.3.8 - A aplicação do algoritmo de Kumamoto & Henley à árvore de eventos \mathcal{A} , definida no Exemplo 2.3.5, fornece o seguinte resultado:

Etapa KH.2 $C_\emptyset = Z = U = \emptyset \quad \mathcal{D}_{KH} = \{ \{ (\emptyset, \{T^c\}, \emptyset) \} \delta_{\prec \emptyset, \emptyset \succ} \}$.

Etapa KH.3 A aplicação do Procedimento Ativa à forma $\{ (\emptyset, \{T^c\}, \emptyset) \} \delta_{\prec \emptyset, \emptyset \succ}$, fornece:

caso 1.

$$\begin{aligned} D &= \{T^c\} & F &= T^c = E_{21}^c \cap E_{22}^c & H &= \{E_{21}^c, E_{22}^c\} & \{ \{T^c\} \} &\equiv \{ \{E_{21}^c, E_{22}^c\} \} \\ A &= \emptyset & B &= \{E_{21}^c, E_{22}^c\} & W &= \emptyset. \end{aligned}$$

Assim, a forma obtida é $\{ (\emptyset, \{E_{21}^c, E_{22}^c\}, \emptyset) \} \delta_{\prec \emptyset, \emptyset \succ}$, que é não-ativa. Então, aplicando o Procedimento Ativa à essa forma, tem-se que:

caso 2.

$$D = \{E_{21}^c, E_{22}^c\} \quad F = E_{21}^c = Q^c \cup E_{32}^c \quad H = \{Q^c, E_{32}^c\} \quad L = \emptyset$$

$$\begin{aligned} \{ \{E_{21}^c, E_{22}^c\} \} &\equiv \{ \{E_{22}^c\} \} \wedge (\{ \{Q^c\} \} \vee \{ \{E_{32}^c\} \}) \\ &= \{ \{Q^c, E_{22}^c\} \} \vee \{ \{E_{22}^c, E_{32}^c\} \} \\ &\equiv (\{ \{Q^c\} \} \wedge \{ \{E_{22}^c\} \}) \vee \{ \{E_{22}^c, E_{32}^c\} \}. \end{aligned}$$

Logo,

$$\begin{array}{lll} A_1 = \{Q^c\} & B_1 = \{E_{22}^c\} & W_1 = \{Q\} \\ A_2 = \emptyset & B_2 = \{E_{22}^c, E_{32}^c\} & W_2 = \emptyset. \end{array}$$

Assim,

$$\mathcal{D}_{KH} = \left\{ \left\{ (\{Q^c\}, \{E_{22}^c\}, \{Q\}), (\emptyset, \{E_{22}^c, E_{32}^c\}, \emptyset) \right\} \delta_{\langle \emptyset, \emptyset \rangle} \right\}.$$

Etapas KH.4 e KH.5 Escolhendo Q^c e realizando as operações

$$(\delta_{\langle \emptyset, \emptyset \rangle} \prec \{Q^c\}, \emptyset \succ) \text{ e } (\delta_{\langle \emptyset, \emptyset \rangle} \prec \emptyset, \{Q^c\} \succ),$$

tem-se as formas

$$\left\{ (\emptyset, \{E_{22}^c, E_{32}^c\}, \emptyset) \right\} \delta_{\langle \{Q^c\}, \emptyset \rangle}$$

e

$$\left\{ (\emptyset, \{E_{22}^c\}, \{Q\}), (\emptyset, \{E_{22}^c, E_{32}^c\}, \emptyset) \right\} \delta_{\langle \emptyset, \{Q^c\} \rangle},$$

que são não-ativas. Logo,

$$\mathcal{D}_{KH} = \left\{ \left\{ (\emptyset, \{E_{22}^c, E_{32}^c\}, \emptyset) \right\} \delta_{\langle \{Q^c\}, \emptyset \rangle}, \left\{ (\emptyset, \{E_{22}^c\}, \{Q\}), (\emptyset, \{E_{22}^c, E_{32}^c\}, \emptyset) \right\} \delta_{\langle \emptyset, \{Q^c\} \rangle} \right\}.$$

Etapa KH.3 Considerando a forma $\left\{ (\emptyset, \{E_{22}^c, E_{32}^c\}, \emptyset) \right\} \delta_{\langle \{Q^c\}, \emptyset \rangle}$, pelo procedimento ativa, tem-se que:

caso 2.

$$D = \{E_{22}^c, E_{32}^c\} \quad F = E_{22}^c = Q \cup R^c \cup S \quad H = \{Q, R^c, S\} \quad L = \{P^c, S^c, R\}.$$

Como $G = E_{32}^c = P^c \cap S^c \cap R$,

$$\begin{aligned} \{\{E_{22}^c, E_{32}^c\}\} &\equiv \{\{P^c, S^c, R\}\} \wedge (\{\{Q\}\} \vee \{\{R^c\}\} \vee \{\{S\}\}) \\ &= \{\{Q, P^c, S^c, R\}\} \vee \{\{R^c, P^c, S^c, R\}\} \vee \{\{S, P^c, S^c, R\}\} \\ &\equiv \{\{Q, P^c, S^c, R\}\}. \end{aligned}$$

Logo,

$$A = \{Q, P^c, S^c, R\} \setminus \{\{Q^c\}, \emptyset\} = \{P^c, S^c, R\} \quad B = \emptyset \quad W = \emptyset.$$

Assim, a forma ativa obtida é:

$$\left\{ (\{P^c, S^c, R\}, \emptyset, \emptyset) \right\} \delta_{\langle \{Q^c\}, \emptyset \rangle}.$$

Por outro lado, considerando a forma

$$\left\{ (\emptyset, \{E_{22}^c\}, \{Q\}), (\emptyset, \{E_{22}^c, E_{32}^c\}, \emptyset) \right\} \delta_{\langle \emptyset, \{Q^c\} \rangle},$$

pelo procedimento ativa, tem-se que:

caso 2.

$$F = E_{22}^c = Q \cup R^c \cup S \quad H = \{Q, R^c, S\}.$$

Para $D_1 = \{E_{22}^c\}$:

$$\begin{aligned} \{\{E_{22}^c\}\} \prec Q, \emptyset \succ &\equiv \{\{Q\}\langle Q, \emptyset \rangle\} \vee \{\{R^c\}\langle Q, \emptyset \rangle\} \vee \{\{S\}\langle Q, \emptyset \rangle\} \\ &\equiv \{\{R^c\}\} \vee \{\{S\}\}. \end{aligned}$$

Para $D_2 = \{E_{22}^c, E_{32}^c\}$, como $G = E_{32}^c = P^c \cap S^c \cap R$,

$$\{\{E_{22}^c, E_{32}^c\}\} \equiv \{\{Q, P^c, S^c, R\}\} \vee \{\{R^c, P^c, S^c, R\}\} \vee \{\{S, P^c, S^c, R\}\}.$$

Logo,

$$(\{\{E_{22}^c\}\} \prec Q, \emptyset \succ) \vee \{\{E_{22}^c, E_{32}^c\}\} \equiv \{\{R^c\}\} \vee \{\{S\}\} \vee \{\{Q, P^c, S^c, R\}\}.$$

Assim,

$$\begin{array}{lll} A_1 = \{R^c\}\langle \emptyset, Q^c \rangle = \{R^c\} & B_1 = \emptyset & W_1 = \emptyset \\ A_2 = \{S\}\langle \emptyset, Q^c \rangle = \{S\} & B_2 = \emptyset & W_2 = \emptyset \\ A_3 = \{Q, P^c, S^c, R\}\langle \emptyset, Q^c \rangle = \{Q, P^c, S^c, R\} & B_3 = \emptyset & W_3 = \emptyset. \end{array}$$

Portanto, a forma obtida é:

$$\left\{ (\{R^c\}, \emptyset, \emptyset), (\{S\}, \emptyset, \emptyset), (\{Q, P^c, S^c, R\}, \emptyset, \emptyset) \right\} \delta_{\langle \emptyset, Q^c \rangle}.$$

Logo,

$$\mathcal{D}_{KH} = \left\{ \left\{ (\{P^c, S^c, R\}, \emptyset, \emptyset) \right\} \delta_{\langle \{Q^c\}, \emptyset \rangle}, \left\{ (\{R^c\}, \emptyset, \emptyset), (\{S\}, \emptyset, \emptyset), (\{Q, P^c, S^c, R\}, \emptyset, \emptyset) \right\} \delta_{\langle \emptyset, \{Q^c\} \rangle} \right\}$$

Etapa KH.5

Escolhendo P^c e realizando as operações

$$(\delta \prec \{Q^c\}, \emptyset \succ) \prec \{P^c\}, \emptyset \succ \text{ e } (\delta \prec \{Q^c\}, \emptyset \succ) \prec \emptyset, \{P^c\} \succ,$$

tem-se a forma

$$\{ (\{\emptyset\}, \emptyset, \emptyset,) \} \delta_{\prec \{Q^c, P^c\}, \emptyset \succ},$$

que é terminal, com $\delta \prec \{Q^c, P^c\}, \emptyset \succ \in [\emptyset]$, e portanto $C_\emptyset = \{\{Q, P\}\}$. E tem-se a forma

$$\{ (\{S^c, R\}, \emptyset, \emptyset,) \} \delta_{\prec \{Q^c\}, \{P^c\} \succ},$$

que é ativa.

Logo, escolhendo S^c e realizando as operações

$$(\delta \prec \{Q^c\}, \{P^c\} \succ) \prec \{S^c\}, \emptyset \succ \text{ e } (\delta \prec \{Q^c\}, \{P^c\} \succ) \prec \emptyset, \{S^c\} \succ,$$

tem-se a forma

$$\{ (\{\emptyset\}, \emptyset, \emptyset,) \} \delta_{\prec \{Q^c, S^c\}, \{P^c\} \succ},$$

que é terminal, com $\delta \prec \{Q^c, S^c\}, \{P^c\} \succ \in [\emptyset]$, e portanto $C_\emptyset = \{\{Q, P\}, \{Q, S\}\}$. E tem-se a forma

$$\{ (\{R\}, \emptyset, \emptyset,) \} \delta_{\prec \{Q^c\}, \{P^c, S^c\} \succ},$$

que é ativa. Portanto, escolhendo R e realizando as operações

$$(\delta \prec \{Q^c\}, \{P^c, S^c\} \succ) \prec \{R\}, \emptyset \succ \text{ e } (\delta \prec \{Q^c\}, \{P^c, S^c\} \succ) \prec \emptyset, \{R\} \succ,$$

tem-se a forma

$$\{ (\{\emptyset\}, \emptyset, \emptyset,) \} \delta_{\prec \{Q^c, R\}, \{P^c, S^c\} \succ},$$

que é terminal, com $\delta \prec \{Q^c, R\}, \{P^c, S^c\} \succ \in [\emptyset]$, e portanto $C_\emptyset = \{\{Q, P\}, \{Q, S\}, \{Q, R^c\}\}$, e tem-se a forma

$$\{ (\{\Omega\}, \emptyset, \emptyset,) \} \delta_{\prec \{Q^c\}, \{P^c, S^c, R\} \succ},$$

que é terminal com $\delta \prec \{Q^c\}, \{P^c, S^c, R\} \succ \in [\Omega]$.

Escolhendo R e realizando as operações

$$(\delta \prec \emptyset, \{Q^c\} \succ) \prec \{R^c\}, \emptyset \succ \text{ e } (\delta \prec \emptyset, \{Q^c\} \succ) \prec \emptyset, \{R^c\} \succ,$$

tem-se a forma

$$\{ (\{\Omega\}, \emptyset, \emptyset), (\{S\}, \emptyset, \emptyset), (\{Q, P^c, S^c, R\}, \emptyset, \emptyset) \} \delta_{\prec \emptyset, \{R^c, Q^c\} \succ},$$

que é terminal, com $\delta_{\prec \emptyset, \{R^c, Q^c\} \succ} \in [\Omega]$, e tem-se a forma

$$\{ (\{S\}, \emptyset, \emptyset), (\{Q, P^c, S^c\}, \emptyset, \emptyset) \} \delta_{\prec \{R^c\}, \{Q^c\} \succ},$$

que é ativa. Assim, escolhendo S e realizando as operações

$$(\delta_{\prec \{R^c\}, \{Q^c\} \succ})_{\prec \{S\}, \emptyset \succ} \text{ e } (\delta_{\prec \{R^c\}, \{Q^c\} \succ})_{\prec \emptyset, \{S\} \succ},$$

tem-se a forma

$$\{ (\{\Omega\}, \emptyset, \emptyset), (\{Q, P^c, S^c\}, \emptyset, \emptyset) \} \delta_{\prec \{R^c\}, \{Q^c, S\} \succ},$$

que é terminal, com $\delta_{\prec \{R^c\}, \{Q^c, S\} \succ} \in [\Omega]$, e tem-se a forma

$$\{ (\{Q, P^c\}, \emptyset, \emptyset) \} \delta_{\prec \{R^c, S\}, \{Q^c\} \succ},$$

que é ativa. Então, escolhendo Q e realizando as operações

$$(\delta_{\prec \{R^c, S\}, \{Q^c\} \succ})_{\prec \{Q\}, \emptyset \succ} \text{ e } (\delta_{\prec \{R^c, S\}, \{Q^c\} \succ})_{\prec \emptyset, \{Q\} \succ},$$

tem-se a forma

$$\{ (\{\emptyset\}, \emptyset, \emptyset) \} \delta_{\prec \{R^c, S, Q\}, \{Q^c\} \succ},$$

que é terminal, com $\delta_{\prec \{R^c, S, Q\}, \{Q^c\} \succ} \in [\emptyset]$, e portanto $C_\emptyset = \{\{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{R, S^c, Q^c\}\}$, e tem-se a forma

$$\{ (\{P^c\}, \emptyset, \emptyset) \} \delta_{\prec \{R^c, S\}, \{Q^c, Q\} \succ},$$

que é ativa. Assim, escolhendo P^c e realizando as operações

$$(\delta_{\prec \{R^c, S\}, \{Q^c, Q\} \succ})_{\prec \{P^c\}, \emptyset \succ} \text{ e } (\delta_{\prec \{R^c, S\}, \{Q^c, Q\} \succ})_{\prec \emptyset, \{P^c\} \succ},$$

tem-se a forma

$$\{ (\{\emptyset\}, \emptyset, \emptyset) \} \delta_{\prec \{R^c, S, P^c\}, \{Q^c, Q\} \succ},$$

que é terminal, com $\delta_{\prec \{R^c, S, P^c\}, \{Q^c, Q\} \succ} \in [\emptyset]$, e portanto $C_\emptyset = \{\{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{R, S^c, Q^c\}, \{R, S^c, P\}\}$, e tem-se a forma

$$\{ (\{\Omega\}, \emptyset, \emptyset) \} \delta_{\prec \{R^c, S\}, \{Q^c, Q, P^c\} \succ},$$

que é terminal, com $\delta_{\prec \{R^c, S\}, \{Q^c, Q, P^c\} \succ} \in [\Omega]$.

Etapa KH.6 Como não existem $A, B \in C_\emptyset$ tal que $A \supset B$, a base completa \mathcal{B} para T é:

$$\mathcal{B} = C_\emptyset = \{ \{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{R, S^c, Q^c\}, \{R, S^c, P\} \}.$$

2.3.3 Cortes Espúrios

O objetivo deste tópico é apresentar as dificuldades que um analista de confiabilidade pode encontrar ao tentar modelar alguns problemas aplicando os métodos convencionais de construção de árvores de eventos, que são, em geral, utilizados na prática. Para tal, será apresentada uma modelagem feita por Fussell, que se encontra em [Lam], para o problema do sistema de iluminação descrito no Exemplo 1.2.3 (Capítulo 1). A árvore de eventos obtida para o evento topo “falta luz” é dada na Figura 2.3.1 abaixo.

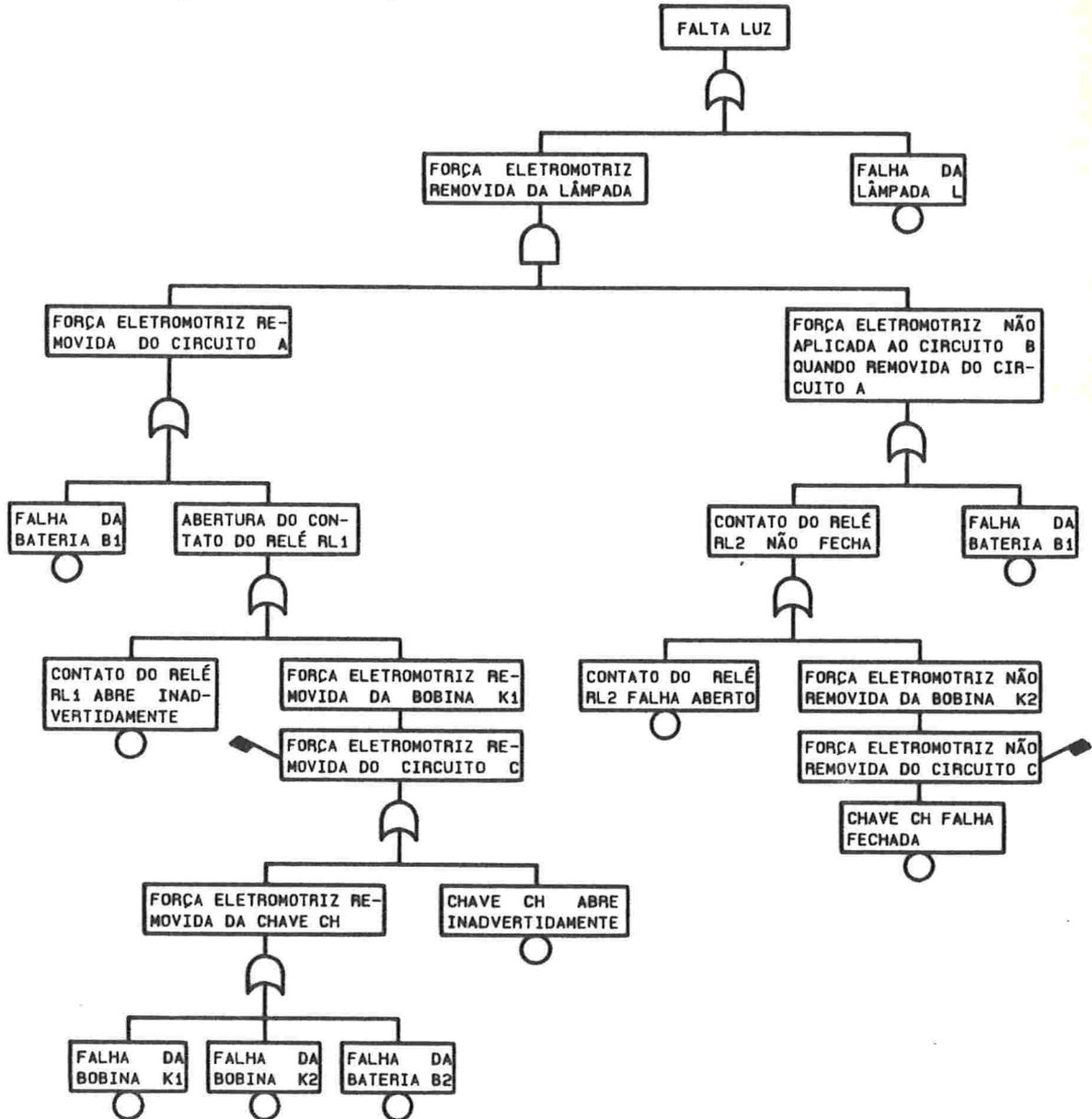


Figura 2.3.1: Árvore de eventos para o evento topo “falta luz”

Nesta árvore existem dois eventos mutuamente exclusivos que estão indicados com bandeiras pretas. Utilizando as definições dadas no Exemplo 2.3.1 para os eventos terminais básicos da árvore, e aplicando o algoritmo DN, a seguinte base completa \mathcal{B} é obtida para o evento topo “falta luz”:

$$\begin{aligned} \mathcal{B} = \{ & \{L\}, \{B1\}, \{RL1_{ai}, RL2_{fa}\}, \\ & \{RL2_{fa}, K1\}, \{RL2_{fa}, K2\}, \{RL2_{fa}, B2\}, \\ & \{RL2_{fa}, CH_{ai}\}, \{RL1_{ai}, CH_{ff}\}, \overbrace{\{K1, CH_{ff}\}}^*, \\ & \overbrace{\{K2, CH_{ff}\}}^*, \overbrace{\{B2, CH_{ff}\}}^*, \overbrace{\{CH_{ai}, CH_{ff}\}}^* \}. \end{aligned}$$

Os cortes mínimos marcados com o símbolo \star são espúrios. O corte mínimo $\{CH_{ai}, CH_{ff}\}$ é obviamente espúrio, pois a chave CH não pode falhar fechada e abrir inadvertidamente ao mesmo tempo. Os outros cortes mínimos espúrios são menos óbvios, mas observe que se $K1$, $K2$ ou $B2$ ocorrerem, para o sistema de iluminação falhar, basta que o evento $RL2_{fa}$ ocorra, sendo a posição da chave CH irrelevante.

2.4 Bases Irredundantes

Quando a base completa \mathcal{B} para o evento topo T , de uma árvore de eventos \mathcal{A} , satisfaz a condição $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) \neq \emptyset$, é possível obter $\mathcal{B}' \subset \mathcal{B}$, tal que $\mathcal{B}' \equiv \mathcal{B}$, e que nenhum $\mathcal{B}'' \subset \mathcal{B}'$ satisfaz $\mathcal{B}'' \equiv \mathcal{B}'$. Uma disjunção normal \mathcal{B}' com estas propriedades é uma base irredundante para T .

Equivalentemente, uma base irredundante é uma disjunção normal, cujos conjuntos suficientes não são supérfluos e nem são formados por eventos terminais básicos supérfluos,

no sentido das definições abaixo.

Definição 2.4.1 *Um conjunto suficiente C é supérfluo em uma disjunção normal δ , se*

$$\delta \equiv \delta - \{C\}.$$

Definição 2.4.2 *Um evento terminal básico E é supérfluo em um conjunto suficiente C , pertencente a uma disjunção normal δ , se*

$$\delta \equiv (\delta - \{C\}) \vee \{C - \{E\}\}.$$

Vários métodos foram propostos para a obtenção das bases irredundantes para um evento topo T , cuja base completa \mathcal{B} satisfaz $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) \neq \emptyset$ (cf. [Qui2], [Reu], [Gha], [Men]). Como a partir destes métodos todas as bases irredundantes são obtidas, a base irredundante mínima para T é simplesmente selecionada deste conjunto. Porém, estes métodos possuem a desvantagem de serem aplicáveis somente á árvores de eventos \mathcal{A} , cujas estruturas não são muito complexas: o mapa de Karnaugh [Men], por exemplo, é um método gráfico, cuja aplicação é restrita à árvores com um número bem reduzido de eventos terminais básicos (6 a 8); os outros métodos, obtém todas as bases irredundantes partindo de duas disjunções normais para T , sendo uma delas a base completa.

Um método bastante simples para a obtenção de uma base irredundante, denominado Teste de Dispensabilidade, que é citado em [Qui1], consiste em:

Obtida a base completa \mathcal{B} para o evento topo T , de uma árvore de eventos \mathcal{A} , $C \in \mathcal{B}$ é dito dispensável, e pode ser eliminado de \mathcal{B} , se

$$(\mathcal{B} - \{C\}) \langle \overline{C}, \emptyset \rangle \equiv \{\{\Omega\}\},$$

onde o operador $\langle \overline{C}, \emptyset \rangle$ é dado pela Definição 2.3.2 (Algoritmo de Kumamoto & Henley). Se C é dispensável, então $\mathcal{B} - \{C\} \equiv \mathcal{B}$. Assim, testa-se um a um os conjuntos suficientes mínimos para T , em \mathcal{B} , para verificar se são dispensáveis ou não. A medida que um conjunto suficiente mínimo dispensável é identificado, este é eliminado de \mathcal{B} , antes de prosseguir os testes. Após todos os conjuntos suficientes mínimos terem sido testados, a base resultante é uma base irredundante para T .

Exemplo 2.4.1 - A aplicação do Teste de Dispensabilidade à árvore de eventos \mathcal{A} , definida no Exemplo 2.3.5, fornece o seguinte resultado:

Pela aplicação dos algoritmos de Quine, Nelson e Kumamoto & Henley à árvore de eventos \mathcal{A} , a base completa para o evento topo T de \mathcal{A} é dada por:

$$\mathcal{B} = \{ \{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{Q^c, R, S^c\}, \{P, R, S^c\} \}.$$

Assim:

O conjunto suficiente mínimo $\{Q, S\}$ não é dispensável, pois:

$$(\mathcal{B} - \{Q, S\}) \langle \{Q^c, S^c\}, \emptyset \rangle \equiv \{ \{P\}, \{R^c\} \}.$$

O conjunto suficiente mínimo $\{Q, R^c\}$ não é dispensável, pois:

$$(\mathcal{B} - \{Q, R^c\}) \langle \{Q^c, R\}, \emptyset \rangle \equiv \{ \{P\}, \{S\} \}.$$

O conjunto suficiente mínimo $\{Q^c, R, S^c\}$ não é dispensável, pois:

$$(\mathcal{B} - \{Q^c, R, S^c\}) \langle \{Q, R^c, S\}, \emptyset \rangle \equiv \{ \{P\} \}.$$

O conjunto suficiente mínimo $\{P, R, S^c\}$ é dispensável, pois:

$$(\mathcal{B} - \{P, R, S^c\}) \langle \{P^c, R^c, S\}, \emptyset \rangle \equiv \{ \{Q\}, \{Q^c\} \} \equiv \{ \{\Omega\} \}.$$

O conjunto suficiente mínimo $\{Q, P\}$ não é dispensável, pois:

$$((\mathcal{B} - \{P, R, S^c\}) - \{Q, P\}) \langle \{Q^c, P^c\}, \emptyset \rangle \equiv \{ \{S\}, \{R^c\} \}.$$

Logo,

$$\{ \{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{Q^c, R, S^c\} \},$$

é a base irredundante obtida através deste algoritmo.

Observação:

- A eliminação de um conjunto suficiente mínimo dispensável, pode fazer outro conjunto suficiente mínimo inicialmente dispensável, indispensável.
- A base irredundante obtida por este método não é necessariamente mínima.

Outro método para a obtenção de uma base irredundante para T é dado em [ACCS]. Este método, proposto por Contini, é um algoritmo simples, que parte de uma disjunção normal δ para T , e obtém uma base irredundante para T , utilizando a identidade de Shannon,

$$\delta \equiv (\{\{E\}\} \wedge \delta \{\{E^c\}, \emptyset\}) \vee (\{\{E^c\}\} \wedge \delta \{\{E\}, \emptyset\}), \quad (2.4.1)$$

onde os operadores $\langle \{E^c\}, \emptyset \rangle$ e $\langle \{E\}, \emptyset \rangle$ são dados pela Definição 2.3.2 (Algoritmo de Kumamoto & Henley).

Apesar de não haver garantias de que a base irredundante mínima é obtida por este algoritmo, o autor comenta, baseado em aplicações à vários exemplos citados na literatura, que se a base irredundante obtida por seu algoritmo não é mínima, ela está bem próxima de uma. O algoritmo é descrito a seguir:

Algoritmo de Contini

O algoritmo de Contini [ACCS] é realizado em duas fases: a primeira, denominada expansão, utiliza (2.4.1), com o objetivo de isolar os possíveis eventos terminais básicos supérfluos; a segunda, denominada redução, utiliza a expressão resultante da fase de expansão e realiza vários testes para eliminar os eventos terminais básicos supérfluos e os conjuntos suficientes supérfluos.

Procedimento de Expansão

O Procedimento de Expansão é realizado através das seguintes etapas:

EXP.1 caso não se disponha de uma disjunção normal δ para o evento topo de interesse, derive uma, através do algoritmo DN ou do algoritmo de Fussell, Henry e Marshall (Seção 2.3.1);

EXP.2 utilizando a identidade de Shannon, obtenha a disjunção

$$(\{\{E\}\} \wedge \delta \{\{E^c\}, \emptyset\}) \vee (\{\{E^c\}\} \wedge \delta \{\{E\}, \emptyset\}) \equiv \delta,$$

onde $(E, E^c) \in \mathcal{C}_\delta = \{(I, I^c) : I, I^c \in (\bigcup \delta) \cap (\overline{\bigcup \delta})\}$ é o par de eventos terminais básicos que aparece o maior número de vezes em δ ;

EXP.3 simplifique $(\bigcup_{C \in \delta} \cap C(\{E^c\}, \emptyset))$ e $(\bigcup_{C \in \delta} \cap C(\{E^c\}, \emptyset))$, usando as identidades

- i) $A \cup (A \cap B) = A$
- ii) $\Omega \cup A = \Omega$
- iii) $A + A^c = \Omega$
- iv) $A + (A^c \cap B) = A \cup B,$

obtendo as expressões $\bigcup_{D \in \delta_1} \cap D$ e $\bigcup_{D \in \delta_2} \cap D$. Chame δ_1 e δ_2 de disjunções normais residuais;

Repita EXP.2 e EXP.3, com cada disjunção normal residual γ obtida no final da etapa EXP.3, até que $C_\gamma = \emptyset$, para cada disjunção normal residual considerada.

As disjunções normais residuais obtidas no final do Procedimento de Expansão são irredundantes mínimas, já que eventos terminais básicos complementares não fazem parte de suas expressões.

Procedimento de Redução

Por construção, uma subexpressão típica da expressão obtida no final do Procedimento de Expansão é do tipo:

$$(\{E\} \wedge \delta_1) \vee (\{E^c\} \wedge \delta_2), \quad (2.4.12)$$

onde δ_1 e δ_2 são disjunções normais residuais e os eventos terminais básicos E , E^c foram utilizados na etapa EXP.2 para a realização de uma expansão.

Para realizar a transformação de (2.4.12) em uma base irredundante, é necessário eliminar os conjuntos suficientes supérfluos e os eventos terminais básicos supérfluos da expressão.

Começando das subexpressões contidas nos parênteses mais internos, até chegar nos parênteses mais externos, da expressão obtida no final do Procedimento de Expansão, cada subexpressão do tipo (2.4.12) é transformada em uma base irredundante. Assim, ao final deste procedimento recursivo, obtêm-se uma base irredundante para T .

1. Eliminação de Eventos Terminais Básicos Supérfluos

A transformação de cada subexpressão do tipo (2.4.12) em uma base irredundante é realizada eliminando-se, primeiramente, os eventos terminais básicos superfluos. Como na expressão (2.4.12), δ_1 e δ_2 já são irredundante mínimas, os únicos candidatos a eventos terminais básicos superfluos são E e E^c . Assim tem-se os seguintes casos:

(1) **Conjuntos suficientes onde ambos, E e E^c , são superfluos**

Estes conjuntos são elementos de $\delta_1 \cap \delta_2$. Observe que se $A \in \delta_1 \cap \delta_2$,

$$(\{\{E\}\} \wedge \{A\}) \vee (\{\{E^c\}\} \wedge \{A\}) \equiv \{A\}.$$

(2) **Conjuntos suficientes onde somente E é superfluo**

Se E é superfluo em um conjunto suficiente $A \cup \{E\}$, $A \in \delta_1 - \delta_2$, tem-se

$$\begin{aligned} & \left(\{\{E\}\} \vee (\{\{E\}\} \wedge (\delta_1 - \{A\})) \vee (\{\{E^c\}\} \wedge \delta_2) \right) \langle \bar{A}, \emptyset \rangle \equiv \\ & \left(\{\{E\}\} \vee (\{\{E^c\}\} \wedge \delta_2) \right) \langle \bar{A}, \emptyset \rangle \equiv \\ & \{\{\Omega\}\}. \end{aligned}$$

Como $E, E^c \notin A \in \delta_1 \cup \delta_2$, estes conjuntos são os elementos $A \in \delta_1 - \delta_2$ tais que

$$\delta_2 \langle \bar{A}, \emptyset \rangle \equiv \{\{\Omega\}\}.$$

(3) **Conjuntos suficientes onde somente E^c é superfluo**

Analogamente ao caso (2), estes conjuntos são os elementos de $A \in \delta_2 - \delta_1$ tais que

$$\delta_1 \langle \bar{A}, \emptyset \rangle \equiv \{\{\Omega\}\}.$$

Denominando a disjunção normal formada pelos conjuntos suficientes obtidos no Caso (1) de Γ_{12} , no Caso (2) de Γ_1 e no Caso (3) de Γ_2 , (2.4.12) pode ser escrita como:

$$(\{\{E\}\} \wedge \xi_1) \vee (\{\{E^c\}\} \wedge \xi_2) \vee \Gamma_1 \vee \Gamma_2 \vee \Gamma_{12}, \quad (2.4.13)$$

onde $\xi_1 = \delta_1 - (\Gamma_1 \vee \Gamma_{12})$ e $\xi_2 = \delta_2 - (\Gamma_2 \vee \Gamma_{12})$.

2. Eliminação dos Conjuntos Suficientes Supérfluos

Após a eliminação dos eventos terminais básicos superfluos, tem início o processo de eliminação dos conjuntos suficientes superfluos. Estes conjuntos são dados pelos seguintes casos:

- (1) $(\{E\} \cup A)$, onde $A \in \xi_1$, é superfluo em (2.4.3) se :

$$\left((\{\{E\}\} \wedge (\xi_1 - \{A\})) \vee (\{\{E^c\}\} \wedge \xi_2) \vee \Gamma_1 \vee \Gamma_2 \vee \Gamma_{12} \right) \langle \overline{\{E\} \cup A}, \emptyset \rangle \equiv \{\{\Omega\}\}.$$

Porém, se $\Gamma_2 = \emptyset$, então

$(\xi_1 - \{A\}) \vee \Gamma_1 \vee \Gamma_2 \vee \Gamma_{12} = (\xi_1 - \{A\}) \vee \Gamma_1 \vee \Gamma_{12} = \delta_1 - \{A\}$, e $(\{E\} \cup A)$ não pode ser superfluo, pois δ_1 é irredundante.

- (2) Analogamente a (1), $(\{E^c\} \cup A)$, onde $A \in \xi_2$, é superfluo em (2.4.3) se :

$$\left((\{\{E\}\} \wedge \xi_1) \vee (\{\{E^c\}\} \wedge (\xi_2 - \{A\})) \vee \Gamma_1 \vee \Gamma_2 \vee \Gamma_{12} \right) \langle \overline{\{E^c\} \cup A}, \emptyset \rangle \equiv \{\{\Omega\}\}.$$

Porém, se $\Gamma_1 = \emptyset$, então

$(\xi_2 - \{A\}) \vee \Gamma_1 \vee \Gamma_2 \vee \Gamma_{12} = (\xi_2 - \{A\}) \vee \Gamma_1 \vee \Gamma_{12} = \delta_2 - \{A\}$, e $(\{E^c\} \cup A)$ não pode ser superfluo, pois δ_2 é irredundante.

- (3) $A \in \Gamma_1$ é superfluo em (2.4.3) se:

$$\begin{aligned} & \left((\{\{E\}\} \wedge \xi_1) \vee (\{\{E^c\}\} \wedge \xi_2) \vee (\Gamma_1 - \{A\}) \vee \Gamma_2 \vee \Gamma_{12} \right) \langle \overline{A}, \emptyset \rangle \equiv \\ & \left((\{\{E\}\} \wedge \xi_1) \vee (\{\{E^c\}\} \wedge \xi_2) \vee (\Gamma_1 - \{A\}) \vee (\{\{E\}\} \vee \{\{E^c\}\}) \wedge (\Gamma_2 \vee \Gamma_{12}) \right) \langle \overline{A}, \emptyset \rangle \equiv \\ & \left((\{\{E\}\} \wedge (\xi_1 \vee \Gamma_2 \vee \Gamma_{12})) \vee (\{\{E^c\}\} \wedge \delta_2) \vee (\Gamma_1 - \{A\}) \right) \langle \overline{A}, \emptyset \rangle \equiv \\ & \{\{\Omega\}\}. \end{aligned}$$

Como $A \in \Gamma_1$, então E é superfluo em $(\{E\} \cup A)$ e vale $\delta_2 \langle \overline{A}, \emptyset \rangle \equiv \{\{\Omega\}\}$. Portanto, basta verificar a equivalência

$$\left(\xi_1 \vee \Gamma_2 \vee \Gamma_{12} \vee (\Gamma_1 - \{A\}) \right) \langle \overline{A}, \emptyset \rangle \equiv \{\{\Omega\}\}.$$

Porém, se $\Gamma_2 = \emptyset$, então

$\delta_1 \vee \Gamma_2 \vee \Gamma_{12} \vee (\Gamma_1 - \{A\}) = \delta_1 \vee (\Gamma_1 - \{A\}) \vee \Gamma_{12} = \delta_1 - \{A\}$, e A não pode ser superfluo, pois δ_1 é irredundante.

(4) Se $A \in \Gamma_2$, então analogamente ao Caso (3), A é supérfluo em (2.4.13) se:

$$\begin{aligned} & \left((\{\{E\}\} \wedge \xi_1) \vee (\{\{E^c\}\} \wedge \xi_2) \vee \Gamma_1 \vee (\Gamma_2 - \{A\}) \vee \Gamma_{12} \right) \langle \bar{A}, \emptyset \rangle \equiv \\ & \left(\xi_2 \vee \Gamma_1 \vee (\Gamma_2 - \{A\}) \vee \Gamma_{12} \right) \langle \bar{A}, \emptyset \rangle = \{\{\Omega\}\}. \end{aligned}$$

Porém, se $\Gamma_1 = \emptyset$, então

$\xi_2 \vee \Gamma_1 \vee \Gamma_{12} \vee (\Gamma_2 - \{A\}) = \xi_2 \vee (\Gamma_2 - \{A\}) \vee \Gamma_{12} = \delta_2 - \{A\}$, e A não pode ser supérfluo, pois δ_2 é irredundante.

(5) Se $A \in \Gamma_{12}$, então nenhuma identidade precisa ser verificada, pois A não pode ser supérfluo, ou seja,

$$\begin{aligned} & \left((\{\{E\}\} \wedge \xi_1) \vee (\{\{E^c\}\} \wedge \xi_2) \vee \Gamma_1 \vee \Gamma_2 \vee (\Gamma_{12} - \{A\}) \right) \langle \bar{A}, \emptyset \rangle \equiv \\ & \left((\{\{E\}\} \wedge (\delta_1 - \{A\})) \vee (\{\{E^c\}\} \wedge (\delta_2 - \{A\})) \right) \langle \bar{A}, \emptyset \rangle \neq \{\{\Omega\}\}, \end{aligned}$$

pois $(\delta_1 - \{A\}) \langle \bar{A}, \emptyset \rangle \neq \Omega$ e $(\delta_2 - \{A\}) \langle \bar{A}, \emptyset \rangle \neq \Omega$, devido a δ_1 e δ_2 serem irredundantes.

Pode-se resumir estes resultados da seguinte forma:

- 1) Se $\Gamma_1 = \emptyset$ e $\Gamma_2 = \emptyset$, a expressão (2.4.13) não possui conjuntos suficientes supérfluos.
- 2) Se $\Gamma_1 = \emptyset$ e $\Gamma_2 \neq \emptyset$, os conjuntos suficientes supérfluos pertencem a ξ_1 .
- 3) Se $\Gamma_1 \neq \emptyset$ e $\Gamma_2 = \emptyset$, os possíveis conjuntos suficientes supérfluos pertencem a ξ_2 .
- 4) Se $\Gamma_1 \neq \emptyset$ e $\Gamma_2 \neq \emptyset$, os possíveis conjuntos suficientes supérfluos podem pertencer a ξ_1 , ξ_2 , Γ_1 ou Γ_2 .

Exemplo 2.4.2 - A aplicação do algoritmo de Contini à árvore de eventos \mathcal{A} , definida no Exemplo 2.3.5, fornece o seguinte resultado:

Procedimento de Expansão:

Etapa EXP.1

Etapa DN.1 Através do método “top-down”:

$$T = E_{21} \cup E_{22}$$

$$\begin{aligned} & (Q \cap E_{32}) \cup (Q^c \cap R \cap S^c) \\ &= [Q \cap (P \cup S \cup R^c)] \cup [Q^c \cap R \cap S^c]. \end{aligned}$$

Etapa DN.2

$$\begin{aligned} T &= [Q \cap (P \cup S \cup R^c)] \cup [Q^c \cap R \cap S^c] \\ &= (Q \cap P) \cup (Q \cap S) \cup (Q \cap R^c) \cup (Q^c \cap R \cap S^c). \end{aligned}$$

Etapa EXP.2

$$\delta = (\{\{Q\}\} \wedge (\{\{P\}\} \vee \{\{S\}\} \vee \{\{R^c\}\})) \vee (\{\{Q^c\}\} \wedge \{\{R, S^c\}\}).$$

Etapa EXP.3

$$\delta_1 = \{ \{P\}, \{S\}, \{R^c\} \}$$

e

$$\delta_2 = \{ \{R, S^c\} \}.$$

Procedimento de Redução:

1. Eliminação de eventos terminais básicos supérfluos.

(1) Como não existem conjuntos suficientes comuns em δ_1 e δ_2 , Q e Q^c não podem ser ambos supérfluos.

(2)

Q não é supérfluo em $\{Q, P\}$, pois

$$(\{\{R, S^c\}\}) \setminus (\{P^c\}, \emptyset) = \{\{R, S^c\}\}.$$

Q não é supérfluo em $\{Q, S\}$, pois

$$(\{\{R, S^c\}\}) \setminus (\{S^c\}, \emptyset) = \{\{\emptyset\}\}.$$

Q não é supérfluo em $\{Q, R^c\}$, pois

$$(\{\{R, S^c\}\}) \setminus (\{R\}, \emptyset) = \{\{\emptyset\}\}.$$

(3)

Q^c não é supérfluo em $\{Q^c, R, S^c\}$, pois

$$(\{\{P\}\} \vee \{\{S\}\} \vee \{\{R^c\}\}) \langle \{R^c, S\}, \emptyset \rangle \equiv \{\{P\}\}$$

2. Eliminação de conjuntos suficientes supérfluos.

Como $\Gamma_1 = \Gamma_2 = \Gamma_{12} = \emptyset$, nenhum conjunto suficiente pertencente a δ é supérfluo. Logo,

$$\delta = \{ \{Q, P\}, \{Q, S\}, \{Q, R^c\}, \{Q^c, R, S^c\} \},$$

é a base irredundante obtida através deste método.

Capítulo 3

Análise Quantitativa

3.1 Introdução

No capítulo anterior considerou-se apenas o aspecto determinístico das relações envolvendo eventos especiais de uma árvore de eventos, tais como eventos terminais básicos, eventos suficientes mínimos e evento topo. Neste capítulo, o interesse está em estudar o comportamento dinâmico desses eventos que, ao longo do tempo, apresentam variações de estado regidas por mecanismos aleatórios.

Suponha que um determinado modo de falha T de um sistema de engenharia é modelado por uma árvore de eventos. Para precaver-se contra os prejuízos ou riscos envolvidos na ocorrência de T , um conjunto de medidas acauteladoras, comumente chamado de programa de manutenção é adotado, com o objetivo de verificar, ao longo do tempo, a ocorrência de eventos terminais básicos indesejáveis, e remover suas causas, quando possível. Este processo de remoção, normalmente constituído por uma ação de reparo ou substituição de um componente do sistema, renova as condições para que esses eventos terminais básicos voltem a ocorrer, fazendo com que esse efeito “on-off”, que se alterna ao longo do tempo, seja propagado, a partir dos eventos terminais básicos, para todos os demais eventos da árvore.

Para avaliar o efeito desse conjunto de medidas acauteladoras, costuma-se estudar o comportamento dos processos que descrevem, para cada evento E de interesse da árvore de eventos, a ausência e a presença, ao longo do tempo, de condições que impliquem sua ocorrência. Nesse estudo, cujo escopo depende do nível de detalhe que se deseja a respeito desses processos e que constitui o que normalmente é denominado análise quantitativa da árvore de eventos, a seguinte terminologia é adotada:

1. O tempo de permanência (“sojourn time”) no estado que representa a presença (ausência) de condições que impliquem a ocorrência de E é denominado tempo de espera (tempo de remoção) para E ;
2. Se em um dado instante, as condições que impliquem a ocorrência de E estão ausentes (presentes) diz-se que E está indisponível (disponível) neste instante.

Na análise quantitativa de uma árvore de eventos, as características do comportamento dinâmico de um evento E de interesse, tipicamente avaliadas, são:

- o número médio de ocorrências do evento E no intervalo $[0, t]$;
- o número médio de remoções do evento E no intervalo $[0, t]$;
- a indisponibilidade do evento E no instante t ;
- a função de densidade do número médio de ocorrências do evento E ;
- a função de densidade do número médio de remoções do evento E ;
- a intensidade condicional de ocorrência do evento E ;
- a intensidade condicional de remoção do evento E ;

Expressões analíticas para as características dinâmicas relacionadas às ocorrências de eventos especiais de uma árvore de eventos, tais como eventos terminais básicos, eventos suficientes mínimos e evento topo, foram desenvolvidas por Vesely [Ves]. Kumamoto & Henley [HK] desenvolveram expressões analíticas para as características dinâmicas relacionadas às remoções de eventos terminais básicos e eventos suficientes mínimos.

Tanto Vesely [Ves] como Kumamoto & Henley, desenvolveram as expressões para o caso de árvores de eventos, cuja base completa \mathcal{B} , para o evento topo, satisfaz a condição $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) = \emptyset$.

Generalizações das expressões desenvolvidas por Vesely foram obtidas por Inagaki & Henley [IH], para o caso de árvores de eventos, cuja base completa \mathcal{B} , para o evento topo, satisfaz a condição $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) \neq \emptyset$.

Em seus trabalhos, Vesely, Kumamoto & Henley e Inagaki & Henley não procuraram utilizar ferramentas formais da teoria dos processos estocásticos, no desenvolvimento das expressões analíticas para as características dinâmicas dos eventos especiais que trataram. As expressões analíticas aparecem em meio a uma discussão informal do problema, dificultando não somente a leitura dos trabalhos como também a percepção dos aspectos fundamentais dos modelos e da necessidade de determinadas hipóteses.

Neste capítulo, procurar-se-á preencher essas lacunas deixadas pelos autores mencionados acima, no contexto de árvores de eventos, cuja base completa \mathcal{B} , para o evento topo, satisfaz $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) \neq \emptyset$. Também serão desenvolvidas para este caso, as expressões analíticas das características dinâmicas relacionadas às remoções de eventos especiais de uma árvore de eventos. Assim como nos trabalhos mencionados, serão tratados apenas eventos especiais como eventos terminais básicos, eventos suficientes mínimos e evento topo.

3.2 Comportamento Dinâmico de Eventos

Nesta seção, serão estudadas, através da construção de processos estocásticos, as características principais do comportamento dinâmico de eventos especiais de uma árvore de eventos \mathcal{A} em (Ω, \mathcal{F}, P) . Os eventos especiais a serem tratados são eventos terminais básicos, eventos suficientes mínimos e evento topo.

3.2.1 Eventos Terminais Básicos

Seja $\{E(t) : t \geq 0\}$, o processo estocástico que descreve o comportamento dinâmico de um evento terminal básico $E \in \mathcal{E}$, ao longo do tempo, onde

$$E(t) = \begin{cases} 1 & \text{se } E \text{ está indisponível no instante } t, \\ 0 & \text{caso contrário.} \end{cases}$$

Uma realização típica deste processo é mostrada na figura 3.2.1 abaixo, onde TF_{Ei} representa o i -ésimo tempo de espera para E e TR_{Ei} representa o i -ésimo tempo de remoção para E .

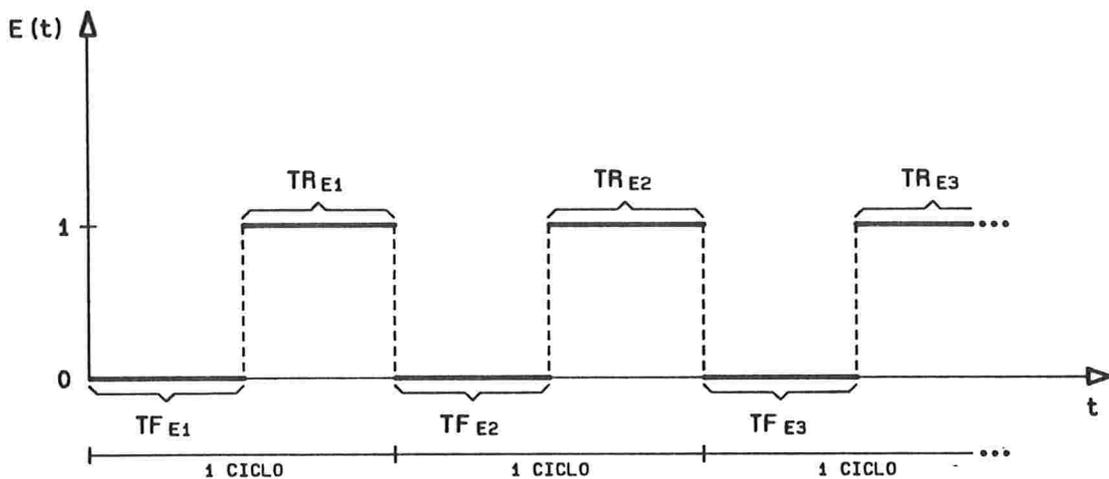


Figura 3.2.1: Realização típica do processo $\{E(t) : t \geq 0\}$

Observações: Para que haja consistência, supõe-se, neste capítulo, que:

- $TF_{E^c i} = TR_{Ei}$,
- $TR_{E^c i} = TF_{Ei}$.

A figura 3.2.1 mostra que o comportamento de $E(t)$, ao longo do tempo, tem características regenerativas, isto é, se repete ao final de cada ciclo. Assim, se os tempos

de duração destes ciclos forem variáveis aleatórias não-negativas, independentes e identicamente distribuídas, o processo de contagem $\{N_E(t) : t \geq 0\}$, do número de ciclos no intervalo $[0, t]$, que coincide com o número de remoções de E , no mesmo intervalo, é um processo de renovação, com função de renovação $M_E(t) = \mathbf{E}[N_E(t)]$.

Em aplicações práticas na engenharia, onde a remoção de um evento terminal básico representa a correção de uma falha detectada em um equipamento, costuma-se fazer esse tipo de suposição, levando-se em conta os efeitos e as características do programa de manutenção a que é submetido o equipamento.

Utilizando então os fundamentos da teoria da renovação, serão demonstradas algumas das características do comportamento dinâmico de E , tais como:

- o número médio de ocorrências do evento E no intervalo $[0, t]$;
- o número médio de remoções do evento E no intervalo $[0, t]$;
- a indisponibilidade do evento E no instante t .

Proposição 3.2.1 *Suponha que as seqüências $\{TF_{Ei} : i \geq 1\}$ e $\{TR_{Ei} : i \geq 1\}$ consistem de variáveis aleatórias independentes e identicamente distribuídas, onde TF_{Ei} , tem função de distribuição F_E e TR_{Ei} , tem função de distribuição G_E , $i = 1, 2, \dots$, com $F_E(0) = G_E(0) = 0$. Então, $\Gamma_E = \{T_{E_n} : n \geq 0\}$, onde $T_{E_0} = 0$ e $T_{E_n} = \sum_{i=1}^n (TF_{Ei} + TR_{Ei})$, é um processo de renovação, com função de distribuição do tempo entre renovações dada por*

$$H_E(t) = P\{T_{E_n} - T_{E_{(n-1)}} \leq t\} = \int_0^t G_E(t-x) dF_E(x), \quad t \geq 0 \quad e \quad n = 1, 2, \dots$$

Nestas condições, se $N_{1E}(t)$ é o número de vezes que E ocorre em $[0, t]$, e $N_{0E}(t)$ é o número de vezes que E é removido em $[0, t]$, então:

a) O número médio de ocorrências de E no intervalo $[0, t]$, $W_E(t) = \mathbf{E}[N_{1E}(t)]$, é dado por

$$W_E(t) = F_E(t) + \int_0^t F_E(t-x) dM_E(x), \quad t \geq 0,$$

b) O número médio de remoções de E no intervalo $[0, t]$, $V_E(t) = \mathbf{E}[N_{0E}(t)]$, é dado por

$$V_E(t) = H_E(t) + \int_0^t H_E(t-x) dM_E(x), \quad t \geq 0,$$

c) A indisponibilidade de E no instante t , $U_E(t)$, definida como $P\{E(t) = 1\}$, $t \geq 0$, é dada por

$$U_E(t) = P\{E(t) = 1\} = F_E(t) - H_E(t) + \int_0^t [F_E(t-x) - H_E(t-x)] dM_E(x), \quad t \geq 0,$$

onde $M_E(t) = \sum_{n=1}^{\infty} P\{T_{E_n} \leq t\} = \sum_{n=1}^{\infty} H_E^{(n)}(t)$ é a função de renovação de Γ_E .

Prova -

a) Como $W_E(t) = \mathbf{E}[N_{1E}(t)]$, condicionando $N_{1E}(t)$ a T_{E1} e aplicando a lei das probabilidades totais, tem-se que

$$\mathbf{E}[N_{1E}(t)/T_{E1} = x] = \begin{cases} P\{TF_{E1} \leq t/T_{E1} = x\} & \text{se } t < x, \\ 1 + \mathbf{E}[N_{1E}(t-x)] & \text{se } t \geq x \end{cases}$$

e

$$\begin{aligned} W_E(t) &= \mathbf{E}[N_{1E}(t)] \\ &= \int_0^{\infty} \mathbf{E}[N_{1E}(t)/T_{E1} = x] dH_E(x) \\ &= \int_0^t [1 + \mathbf{E}[N_{1E}(t-x)]] dH_E(x) + \int_t^{\infty} P\{TF_{E1} \leq t/T_{E1} = x\} dH_E(x) \\ &= P\{TF_{E1} \leq t, TF_{E1} + TR_{E1} > t\} + P\{TF_{E1} + TR_{E1} \leq t\} \\ &\quad + \int_0^t \mathbf{E}[N_{1E}(t-x)] dH_E(x) \\ &= P\{TF_{E1} \leq t\} + \int_0^t \mathbf{E}[N_{1E}(t-x)] dH_E(x) \\ &= F_E(t) + \int_0^t \mathbf{E}[N_{1E}(t-x)] dH_E(x). \end{aligned}$$

Assim, $W_E(t) = \mathbf{E}[N_{1E}(t)]$ satisfaz uma equação de renovação, cuja solução (ver Karlin & Taylor [KT]) é

$$W_E(t) = F_E(t) + \int_0^t F_E(t-x) dM_E(x), \quad t \geq 0.$$

b) Como $V_E(t) = \mathbf{E}[N_{0E}(t)]$, condicionando $N_{0E}(t)$ a T_{E1} e aplicando a lei das probabilidades totais, tem-se que

$$\mathbf{E}[N_{0E}(t)/T_{E1} = x] = \begin{cases} 0 & \text{se } t < x, \\ 1 + \mathbf{E}[N_{0E}(t-x)] & \text{se } t \geq x, \end{cases}$$

e

$$\begin{aligned} V_E(t) &= \mathbf{E}[N_{0E}(t)] \\ &= \int_0^\infty \mathbf{E}[N_{0E}(t)/T_{E1} = x] dH_E(x) \\ &= \int_0^t [1 + \mathbf{E}[N_{0E}(t-x)]] dH_E(x) \\ &= H_E(t) + \int_0^t \mathbf{E}[N_{0E}(t-x)] dH_E(x). \end{aligned}$$

Portanto, $V_E(t) = \mathbf{E}[N_{0E}(t)]$ satisfaz uma equação de renovação, cuja solução (ver Karlin & Taylor [KT]) é

$$V_E(t) = H_E(t) + \int_0^t H_E(t-x) dM_E(x), \quad t \geq 0.$$

c) Condicionando $\mathbb{1}\{E(t) = 1\}$ a T_{E1} e aplicando a lei das probabilidades totais tem-se que

$$P\{E(t) = 1/T_{E1} = x\} = \begin{cases} P\{TF_{E1} \leq t/T_{E1} = x\} & \text{se } t < x, \\ P\{E(t-x) = 1\} & \text{se } t \geq x, \end{cases}$$

e

$$\begin{aligned}
 U_E(t) &= P\{E(t) = 1\} \\
 &= \int_0^\infty P\{E(t) = 1 / T_{E1} = x\} dH_E(x) \\
 &= \int_0^t U_E(t-x) dH_E(x) + \int_t^\infty P\{TF_{E1} \leq t / T_{E1} = x\} dH_E(x) \\
 &= \int_0^t U_E(t-x) dH_E(x) + P\{TF_{E1} \leq t, TF_{E1} + TR_{E1} > t\} \\
 &= \int_0^t U_E(t-x) dH_E(x) + P\{TF_{E1} + TR_{E1} > t\} - P\{TF_{E1} > t\} \\
 &= F_E(t) - H_E(t) + \int_0^t U_E(t-x) dH_E(x).
 \end{aligned}$$

Assim, $U_E(t) = P\{E(t) = 1\}$ satisfaz uma equação de renovação, cuja solução (ver Karlin & Taylor [KT]) é

$$U_E(t) = F_E(t) - H_E(t) + \int_0^t [F_E(t-x) - H_E(t-x)] dM_E(x), \quad t \geq 0.$$

Observação:

- O mesmo resultado pode ser obtido notando que

$$E(t) = N_{1E}(t) - N_{0E}(t) = \begin{cases} 1 & \text{se } E \text{ está indisponível no instante } t, \\ 0 & \text{caso contrário.} \end{cases}$$

Portanto, $P\{E(t) = 1\} = P\{N_{1E}(t) - N_{0E}(t) = 1\} = \mathbf{E}[N_{1E}(t)] - \mathbf{E}[N_{0E}(t)]$.

Basta substituir a expressão acima pelos resultados obtidos em a) e b).

□

Observações: Tendo em vista as hipóteses de consistência estabelecidas nas observações que seguem a Figura 3.2.1:

- $W_{E^c}(t) = V_E(t)$,
- $V_{E^c}(t) = W_E(t)$,
- $U_{E^c}(t) = 1 - U_E(t)$.

Corolário 3.2.1 *Nas condições da Proposição 3.2.1, se F_E e G_E são absolutamente contínuas com funções de densidade f_E e g_E , respectivamente, então w_E , a função de densidade do número médio de ocorrências de E , e v_E , a função de densidade do número médio de remoções de E , satisfazem ao seguinte sistema de equações simultâneas:*

$$\begin{cases} w_E(t) = f_E(t) + \int_0^t f_E(t-x)v_E(x)dx \\ v_E(t) = \int_0^t g_E(t-u)w_E(u)du. \end{cases} \quad (3.2.1)$$

Além disso, se $\frac{d\mathbf{E}[N_{kE}(t)]}{dt} = z_{kE}(t)$, então

$$\lim_{h \downarrow 0} \frac{1}{h} P\{E(t) = 1 - k, E(t+h) = k\} = z_{kE}(t). \quad (3.2.2)$$

Mais precisamente,

$$\lim_{h \downarrow 0} \frac{1}{h} \int_0^t P\{E(u) = 1 - k, E(u+h) = k\} du = \mathbf{E}[N_{kE}(t)] = Z_{kE}(t), \quad (3.2.3)$$

onde

$$Z_{kE}(t) = \begin{cases} W_E(t) & \text{se } k = 1, \\ V_E(t) & \text{se } k = 0, \end{cases}$$

e

$$z_{kE}(t) = \begin{cases} w_E(t) & \text{se } k = 1, \\ v_E(t) & \text{se } k = 0, \end{cases}$$

$k = 0, 1$, $t \geq 0$.

Prova -

(3.2.1) Da proposição 3.2.1 tem-se que

$$\begin{aligned} W_E(t) &= F_E(t) + \int_0^t F_E(t-x) dM_E(x) \\ &= F_E(t) + \int_0^t F_E(t-x) m_E(x) dx \end{aligned}$$

e

$$\begin{aligned} V_E(t) &= H_E(t) + \int_0^t H_E(t-x) dM_E(x) \\ &= H_E(t) + \int_0^t H_E(t-x) m_E(x) dx, \end{aligned}$$

onde $m_E(t) = \frac{dM_E(t)}{dt}$. Assim,

$$\begin{aligned} \frac{dW_E(t)}{dt} &= w_E(t) \\ &= f_E(t) + \int_0^t f_E(t-x) m_E(x) dx + F_E(0) m_E(t) \\ &= f_E(t) + \int_0^t f_E(t-x) m_E(x) dx \end{aligned}$$

e

$$\begin{aligned} \frac{dV_E(t)}{dt} &= v_E(t) \\ &= h_E(t) + \int_0^t h_E(t-x) m_E(x) dx + H_E(0) m_E(t) \\ &= h_E(t) + \int_0^t h_E(t-x) m_E(x) dx. \end{aligned}$$

Como $M_E(t) = V_E(t)$, tem-se que

$$w_E(t) = f_E(t) + \int_0^t f_E(t-x) v_E(x) dx$$

e

$$v_E(t) = h_E(t) + \int_0^t h_E(t-x) v_E(x) dx.$$

Mas,

$$\begin{aligned} \int_0^t h_E(t-x)v_E(x)dx &= \int_0^t \left(\int_0^{t-x} g_E(t-x-s)f_E(s)ds \right) v_E(x)dx \\ &= \int_0^t \left(\int_x^t g_E(t-u)f_E(u-x)du \right) v_E(x)dx \\ &= \int_0^t g_E(t-u) \left(\int_0^u f_E(u-x)v_E(x)dx \right) du. \end{aligned}$$

Portanto,

$$\begin{aligned} v_E(t) &= h_E(t) + \int_0^t h_E(t-x)v_E(x)dx \\ &= \int_0^t g_E(t-u)f_E(u)du + \int_0^t g_E(t-u) \int_0^u f_E(u-x)v_E(x)dx du \\ &= \int_0^t g_E(t-u) \left[f_E(u) + \int_0^u f_E(u-x)v_E(x)dx \right] du \\ &= \int_0^t g_E(t-u)w_E(u)du. \end{aligned}$$

(3.2.2) Segue imediatamente de (3.2.3).

(3.2.3) Fixados $\omega \in \Omega$ e $\epsilon > 0$ arbitrário, existe $\tau > 0$, tal que para todo $0 < h < \tau$ valem:

- $|N_{kE}(t+h, \omega) - N_{kE}(t, \omega)| < \epsilon$ (pois $E(t)$ tem trajetórias contínuas à direita quase certamente)

e

- $N_{kE}(t, \omega) - \epsilon < N_{kE}(t, \omega) \leq$

$$\frac{1}{h} \int_0^t \mathbb{1}\{E(u, \omega) = 1 - k, E(u+h, \omega) = k\} du \leq$$

$$N_{kE}(t+h, \omega) < N_{kE}(t+h, \omega) + \epsilon$$

$$\text{(pois } hN_{kE}(t, \omega) \leq \int_0^t \mathbb{1}\{E(u, \omega) = 1 - k, E(u+h, \omega) = k\} du \leq hN_{kE}(t+h, \omega)\text{)}.$$

Portanto, o limite

$$\lim_{h \downarrow 0} \frac{1}{h} \int_0^t \mathbb{1}\{E(u, \omega) = 1 - k, E(u+h, \omega) = k\} du$$

está definido quase certamente, e é igual a $N_{kE}(t)$, $k = 0, 1$.

Por outro lado, como

$$\left| \frac{1}{h} \int_0^t \mathbb{1}\{E(u) = 1 - k, E(u + h) = k\} du \right| \leq t < +\infty$$

e

$$\lim_{h \downarrow 0} \frac{1}{h} \int_0^t \mathbb{1}\{E(u) = 1 - k, E(u + h) = k\} du = N_{kE}(t),$$

pelo teorema da convergência dominada, tem-se que

$$\mathbf{E} \left[\lim_{h \downarrow 0} \frac{1}{h} \int_0^t \mathbb{1}\{E(u) = 1 - k, E(u + h) = k\} du \right] =$$

$$\lim_{h \downarrow 0} \frac{1}{h} \mathbf{E} \left[\int_0^t \mathbb{1}\{E(u) = 1 - k, E(u + h) = k\} du \right] =$$

$$\lim_{h \downarrow 0} \frac{1}{h} \int_0^t P\{E(u) = 1 - k, E(u + h) = k\} du =$$

$$\mathbf{E} [N_{kE}(t)],$$

$$k = 0, 1, \quad t \geq 0.$$

□

Observações: Devido as hipóteses de consistência, não é difícil ver que:

- $w_{E^c}(t) = v_E(t)$,
- $v_{E^c}(t) = w_E(t)$.

Com base nos resultados do Corolário 3.2.1, pode-se estabelecer as características do comportamento dinâmico de E definidas a seguir.

Intensidade condicional de ocorrência de E :

$$\Lambda_E(t) = \lim_{h \downarrow 0} P\{E(t + h) = 1 / E(t) = 0\} = \frac{w_E(t)}{1 - U_E(t)} \quad \text{se } t \geq 0 \text{ e } U_E(t) < 1. \quad (3.2.4)$$

Intensidade condicional de remoção de E :

$$\Upsilon_E(t) = \lim_{h \downarrow 0} \frac{P\{E(t+h) = 0 / E(t) = 1\}}{h} = \frac{v_E(t)}{U_E(t)} \quad \text{se } t \geq 0 \text{ e } U_E(t) > 0. \quad (3.2.5)$$

As intensidades (3.2.4) e (3.2.5) podem ser interpretadas da seguinte forma: para h suficientemente pequeno, $\Lambda_E(t)h$ ($\Upsilon_E(t)h$) é aproximadamente igual a probabilidade de que E , estando disponível (indisponível) no instante t , ocorra (seja removido) antes de h unidades de tempo.

Observações:

- No caso em que E permanece indisponível após sua primeira ocorrência valem:

i) $W_E(t) = U_E(t) = F_E(t)$,

ii) $V_E(t) = v_E(t) = 0$,

iii) $w_E(t) = f_E(t)$,

iv) $\Lambda_E(t) = \frac{f_E(t)}{1 - F_E(t)} \quad \text{se } F_E(t) < 1$,

v) $\Upsilon_E(t) = 0$,

$t \geq 0$.

Neste caso, $\Lambda_E(t)$ está associada a distribuição para a primeira ocorrência de E e é conhecida como intensidade de ocorrência de E .

Se F_E e G_E possuem distribuições exponenciais, pode-se facilmente obter as características dinâmicas de E , utilizando as transformadas de Laplace. Estes resultados podem ser encontrados em Henley & Kumamoto [HK] e estão resumidos no seguinte corolário:

Corolário 3.2.2 *Sejam*

$$f_E(t) = \lambda_E \exp\{-\lambda_E t\}, \quad \lambda_E > 0, \quad t \geq 0$$

e

$$g_E(t) = \mu_E \exp\{-\mu_E t\}, \quad \mu_E > 0, \quad t \geq 0.$$

Então

$$\begin{aligned} w_E(t) &= \frac{\lambda_E \mu_E}{\lambda_E + \mu_E} + \frac{\lambda_E^2}{\lambda_E + \mu_E} \exp\{-(\lambda_E + \mu_E)t\}, \\ v_E(t) &= \frac{\lambda_E \mu_E}{\lambda_E + \mu_E} - \frac{\lambda_E \mu_E}{\lambda_E + \mu_E} \exp\{-(\lambda_E + \mu_E)t\}, \\ W_E(t) &= \frac{\lambda_E \mu_E}{\lambda_E + \mu_E} t + \frac{\lambda_E^2}{(\lambda_E + \mu_E)^2} (1 - \exp\{-(\lambda_E + \mu_E)t\}), \\ V_E(t) &= \frac{\lambda_E \mu_E}{\lambda_E + \mu_E} t - \frac{\lambda_E \mu_E}{(\lambda_E + \mu_E)^2} (1 - \exp\{-(\lambda_E + \mu_E)t\}), \\ U_E(t) &= \frac{\lambda_E}{\lambda_E + \mu_E} (1 - \exp\{-(\lambda_E + \mu_E)t\}). \end{aligned}$$

□

3.2.2 Eventos Suficientes Mínimos

Como um evento suficiente mínimo é da forma $\cap C$, $C \subset \mathcal{E}$ tal que $C \cap \bar{C} = \emptyset$, pode-se descrever o comportamento dinâmico de $S = \cap C$, ao longo do tempo, definindo o processo estocástico $\{S(t) : t \geq 0\}$, onde

$$S(t) = \prod_{E \in C} E(t) = \begin{cases} 1 & \text{se } S \text{ está indisponível no instante } t, \\ 0 & \text{caso contrário.} \end{cases}$$

Normalmente, é razoável admitir-se que os eventos terminais básicos pertencentes a \mathcal{E}_A são independentes. Então, admitindo esta hipótese, que será mantida até que outra suposição seja considerada, as características do comportamento dinâmico de $S = \cap C$ são dadas pelas seguintes proposições:

Proposição 3.2.2 *A indisponibilidade de $S = \cap C$ no instante t , definida como $U_S^*(t) = P\{S(t) = 1\}$, $t \geq 0$, é dada por*

$$U_S^*(t) = P\left\{\prod_{E \in C} E(t) = 1\right\}$$

$$\begin{aligned}
 &= \prod_{E \in C} P\{E(t) = 1\} \\
 &= \prod_{E \in C} U_E(t).
 \end{aligned}$$

Prova - É uma consequência imediata da independência e da definição de $S(t)$. Basta notar que $\{\prod_{E \in C} E(t) = 1\} \Leftrightarrow \{E(t) = 1\}$, para todo $E \in C$.

□

Lema 3.2.1 $\lim_{h \downarrow 0} P\{E(t+h) = 1\} = P\{E(t) = 1\}$.

Prova - Como $\lim_{h \downarrow 0} E(t+h) = E(t)$ (pois $E(t)$ tem trajetórias contínuas à direita quase certamente) e $|E(t+h)| \leq 1$, segue, do teorema da convergência dominada, que

$$\lim_{h \downarrow 0} P\{E(t+h) = 1\} = \lim_{h \downarrow 0} \mathbf{E}[E(t+h)] = \mathbf{E}[\lim_{h \downarrow 0} E(t+h)] = \mathbf{E}[E(t)] = P\{E(t) = 1\}.$$

□

Proposição 3.2.3

1. A função de densidade do número médio de ocorrências de $S = \cap C$ é dada por

$$w_S^*(t) = \sum_{E \in C} w_E(t) \prod_{I \in C - \{E\}} U_I(t), \quad t \geq 0.$$

2. A função de densidade do número médio de remoções de $S = \cap C$ é dada por

$$v_S^*(t) = \sum_{E \in C} v_E(t) \prod_{I \in C - \{E\}} U_I(t), \quad t \geq 0.$$

Prova - Da mesma forma que no Corolário 3.2.1,

$$w_S^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{S(t) = 0, S(t+h) = 1\}$$

e

$$v_S^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{S(t) = 1, S(t+h) = 0\}.$$

Além disso, pelo mesmo Corolário 3.2.1,

$$P\{E(t) = 0, E(t+h) = 1\} = w_E(t)h + o(h)$$

e

$$P\{E(t) = 1, E(t+h) = 0\} = v_E(t)h + o(h),$$

e pelo Lema 3.2.1,

$$\lim_{h \downarrow 0} P\{E(t+h) = 1\} = P\{E(t) = 1\}.$$

Assim,

$$1. \quad P\{S(t) = 0, S(t+h) = 1\} = P\left\{ \prod_{E \in C} E(t) = 0, \prod_{E \in C} E(t+h) = 1 \right\} =$$

$$P\left\{ \bigcup_{E \in C} \left\{ \{E(t) = 0\} \cap \left\{ \bigcap_{E \in C} \{E(t+h) = 1\} \right\} \right\} \right\} =$$

$$\sum_{r=1}^{|C|} (-1)^{r+1} \sum_{\substack{C' \subseteq C \\ |C'|=r}} P\left\{ \left\{ \bigcap_{E \in C'} \{E(t) = 0, E(t+h) = 1\} \right\} \cap \left\{ \bigcap_{I \in (C-C')} \{I(t+h) = 1\} \right\} \right\} =$$

$$\sum_{r=1}^{|C|} (-1)^{r+1} \sum_{\substack{C' \subseteq C \\ |C'|=r}} \prod_{E \in C'} P\{E(t) = 0, E(t+h) = 1\} \prod_{I \in (C-C')} P\{I(t+h) = 1\} =$$

$$\sum_{r=1}^{|C|} (-1)^{r+1} \sum_{\substack{C' \subseteq C \\ |C'|=r}} \prod_{E \in C'} [w_E(t)h + o(h)] \prod_{I \in (C-C')} P\{I(t+h) = 1\}.$$

Portanto,

$$w_S^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{S(t) = 0, S(t+h) = 1\} =$$

$$\sum_{r=1}^{|C|} (-1)^{r+1} \sum_{\substack{C' \subseteq C \\ |C'|=r}} \lim_{h \downarrow 0} \frac{1}{h} \left[\prod_{E \in C'} [w_E(t)h + o(h)] \prod_{I \in (C-C')} P\{I(t+h) = 1\} \right] =$$

$$\sum_{E \in C} w_E(t) \prod_{I \in C - \{E\}} P\{I(t) = 1\} =$$

$$\sum_{E \in C} w_E(t) \prod_{I \in C - \{E\}} U_I(t).$$

2. De forma análoga a 1. ,

$$P\{S(t) = 1, S(t+h) = 0\} = P\left\{ \prod_{E \in C} E(t) = 1, \prod_{E \in C} E(t+h) = 0 \right\} =$$

$$\sum_{r=1}^{|C|} (-1)^{r+1} \sum_{\substack{C' \subseteq C \\ |C'|=r}} \prod_{E \in C'} P\{E(t) = 1, E(t+h) = 0\} \prod_{I \in (C-C')} P\{I(t) = 1\} =$$

$$\sum_{r=1}^{|C|} (-1)^{r+1} \sum_{\substack{C' \subseteq C \\ |C'|=r}} \prod_{E \in C'} [v_E(t)h + o(h)] \prod_{I \in (C-C')} P\{I(t) = 1\}.$$

Portanto,

$$v_S^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{S(t) = 1, S(t+h) = 0\} =$$

$$\sum_{E \in C} v_E(t) \prod_{I \in C - \{E\}} P\{I(t) = 1\} =$$

$$\sum_{E \in C} v_E(t) \prod_{I \in C - \{E\}} U_I(t).$$

□

Obtidos $U_S^*(t)$, $w_S^*(t)$ e $v_S^*(t)$ e observando que os resultados do Corolário 3.2.1 também valem considerando S ao invés de E , pode-se estabelecer as demais características do comportamento dinâmico de S , onde as interpretações são análogas às realizadas no caso de $E \in \mathcal{E}$. Especificamente:

Intensidade condicional de ocorrência de S :

$$\Lambda_S^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{S(t+h) = 1/S(t) = 0\} = \frac{w_S^*(t)}{1 - U_S^*(t)} \quad \text{se } t \geq 0 \text{ e } U_S^*(t) < 1.$$

Intensidade condicional de remoção de S :

$$\Upsilon_S^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{S(t+h) = 0/S(t) = 1\} = \frac{v_S^*(t)}{U_S^*(t)} \quad \text{se } t \geq 0 \text{ e } U_S^*(t) > 0.$$

Número médio de ocorrências de S em $[0, t]$:

$$W_S^*(t) = \int_0^t w_S^*(u) du, \quad t \geq 0.$$

Número médio de remoções de S em $[0, t]$:

$$V_S^*(t) = \int_0^t v_S^*(u) du, \quad t \geq 0.$$

Observações:

- No caso onde, para todo $E \in C$, E permanece indisponível após sua primeira ocorrência valem:

$$(i) \quad W_S^*(t) = U_S^*(t) = \prod_{E \in C} F_E(t),$$

$$(ii) \quad V_S^*(t) = v_S^*(t) = 0,$$

$$(iii) \quad w_S^*(t) = \sum_{E \in C} f_E(t) \prod_{I \in C - \{E\}} F_I(t),$$

$$(iv) \quad \Lambda_S^*(t) = \frac{\sum_{E \in C} f_E(t) \prod_{I \in C - \{E\}} F_I(t)}{1 - \prod_{E \in C} F_E(t)},$$

$$(v) \Upsilon^*_S(t) = 0.$$

- Em [HK], Henley & Kumamoto citam (sem demonstrar) que a expressão para $v_S^*(t)$, no caso onde $(\cup B) \cap (\overline{\cup B}) = \emptyset$ seria:

$$v_S^*(t) = \sum_{E \in C} v_E(t) \prod_{I \in C - \{E\}} (1 - U_I(t)).$$

Como pode-se verificar, através da expressão obtida na Proposição 3.2.3, esta expressão não está correta. A expressão correta é:

$$v_S^*(t) = \sum_{E \in C} v_E(t) \prod_{I \in C - \{E\}} U_I(t).$$

3.2.3 Evento Topo

Seja $\{T(t) : t \geq 0\}$ o processo estocástico que descreve o comportamento dinâmico do evento topo T , ao longo do tempo, onde

$$T(t) = \begin{cases} 1 & \text{se } T \text{ está indisponível no instante } t, \\ 0 & \text{caso contrário.} \end{cases}$$

Como visto no capítulo 2, várias expressões equivalentes podem ser utilizadas para representar o evento topo T . Assim, utilizando uma base \mathcal{B} para T , pode-se escrever

$$T(t) = \bigsqcup_{C \in \mathcal{B}} \prod_{E \in C} E(t) \tag{3.2.6}$$

$$= \max_{C \in \mathcal{B}} \min_{E \in C} E(t). \tag{3.2.7}$$

Já utilizando um recobrimento \mathcal{R} para T , pode-se escrever

$$T(t) = \prod_{D \in \mathcal{R}} \prod_{E \in D} E(t) \tag{3.2.8}$$

$$= \min_{D \in \mathcal{R}} \max_{E \in D} E(t). \tag{3.2.9}$$

As características do comportamento dinâmico de T , utilizando uma das representações (3.2.6) a (3.2.9) para $T(t)$, são estabelecidas a seguir.

Função de Densidade do Número Médio de Ocorrências/Remoções do Evento Topo T

A função de densidade do número médio de ocorrências do evento topo T , $w_T^\circ(t)$, e a função de densidade do número médio de remoções do evento topo T , $v_T^\circ(t)$, podem ser obtidas pelo princípio da inclusão-exclusão, utilizando resultados demonstrados anteriormente. Especificamente, tem-se:

Teorema 3.2.1

1. A função de densidade do número médio de ocorrências do evento topo T , $w_T^\circ(t)$, é dada por

$$w_T^\circ(t) = w_T^\circ(t) - w_T^\#(t), \quad t \geq 0,$$

onde

$$w_T^\circ(t) = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} Z_{\mathcal{B}'},$$

onde

$$Z_{\mathcal{B}'} = \begin{cases} \sum_{E \in \mathcal{Z}} w_E(t) \prod_{I \in \cup \mathcal{B}' - \{E\}} U_I(t) & \text{se } \cap \mathcal{B}' \neq \emptyset, \text{ onde} \\ & \mathcal{Z} = \{E \in \cap \mathcal{B}' : (\cup \mathcal{B}' - \{E\}) \cap \overline{(\cup \mathcal{B}' - \{E\})} = \emptyset\}, \\ 0 & \text{caso contrário.} \end{cases}$$

e

$$w_T^\#(t) = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} \sum_{q=1}^{|\mathcal{B}-\mathcal{B}'|} (-1)^{q+1} \sum_{\substack{\gamma \subseteq (\mathcal{B}-\mathcal{B}') \\ |\gamma|=q}} Z_{\mathcal{B}', \gamma},$$

onde

$$Z_{\mathcal{B}', \gamma} = \begin{cases} \sum_{E \in \mathcal{Z}'} w_E(t) \prod_{I \in (\cup \mathcal{B}' - \{E\})} U_I(t) \prod_{L \in \cup \gamma} U_L(t) & \text{se } \cap \mathcal{B}' \neq \emptyset, \text{ onde} \\ & \mathcal{Z}' = \{E \in \cap \mathcal{B}' : \alpha \cap \bar{\alpha} = \emptyset, \\ & \beta \cap \bar{\beta} = \emptyset, \alpha \cap \bar{\beta} = \emptyset, \\ & \alpha, \beta = \cup \mathcal{B}' - \{E\}, \cup \gamma\}, \\ 0 & \text{caso contrário.} \end{cases}$$

2. A função de densidade do número médio de remoções do evento topo T , $v_T^*(t)$, é dada por

$$v_T^*(t) = v_T^o(t) - v_T^\#(t), \quad t \geq 0,$$

onde

$$v_T^o(t) = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} Y_{\mathcal{B}'},$$

onde

$$Y_{\mathcal{B}'} = \begin{cases} \sum_{E \in \mathcal{Z}} v_E(t) \prod_{I \in \cup \mathcal{B}' - \{E\}} U_I(t) & \text{se } \cap \mathcal{B}' \neq \emptyset \\ 0 & \text{caso contrário.} \end{cases}$$

e

$$v_T^\#(t) = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} \sum_{q=1}^{|\mathcal{B}-\mathcal{B}'|} (-1)^{q+1} \sum_{\substack{\gamma \subseteq (\mathcal{B}-\mathcal{B}') \\ |\gamma|=q}} Y_{\mathcal{B}', \gamma},$$

onde

$$Y_{\mathcal{B}', \gamma} = \begin{cases} \sum_{E \in \mathcal{Z}'} v_E(t) \prod_{I \in (\cup \mathcal{B}' - \{E\})} U_I(t) \prod_{L \in \cup \gamma} U_L(t) & \text{se } \cap \mathcal{B}' \neq \emptyset \\ 0 & \text{caso contrário.} \end{cases}$$

Prova -

1. Assim como no Corolário 3.2.1,

$$w_T^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{T(t) = 0, T(t+h) = 1\}.$$

Por outro lado,

$$P\{T(t) = 0, T(t+h) = 1\} =$$

$$P\left\{ \prod_{C \in \mathcal{B}} \prod_{E \in C} E(t) = 0, \prod_{C \in \mathcal{B}} \prod_{E \in C} E(t+h) = 1 \right\} =$$

$$P\left\{ \bigcup_{C \in \mathcal{B}} \left\{ \left\{ \bigcap_{D \in \mathcal{B}} \left\{ \prod_{E \in D} E(t) = 0 \right\} \right\} \cap \left\{ \prod_{E \in C} E(t+h) = 1 \right\} \right\} \right\} =$$

$$\sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \left\{ \bigcap_{C \in \mathcal{B}'} \left\{ \prod_{E \in C} E(t) = 0, \prod_{E \in C} E(t+h) = 1 \right\} \right\} \cap \left\{ \bigcap_{D \in (\mathcal{B} - \mathcal{B}')} \left\{ \prod_{E \in D} E(t) = 0 \right\} \right\} \right\} = A - B,$$

onde

$$A = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \bigcap_{C \in \mathcal{B}'} \left\{ \prod_{E \in C} E(t) = 0, \prod_{E \in C} E(t+h) = 1 \right\} \right\}$$

e

$$B = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \left\{ \bigcap_{C \in \mathcal{B}'} \left\{ \prod_{E \in C} E(t) = 0, \prod_{E \in C} E(t+h) = 1 \right\} \right\} \cap \left\{ \bigcup_{D \in (\mathcal{B} - \mathcal{B}')} \left\{ \prod_{E \in D} E(t) = 1 \right\} \right\} \right\}.$$

Como

$$\begin{aligned} A &= \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \bigcap_{C \in \mathcal{B}'} \left\{ \prod_{E \in C} E(t) = 0, \prod_{E \in C} E(t+h) = 1 \right\} \right\} = \\ &= \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \bigcap_{C \in \mathcal{B}'} \left\{ \left\{ \bigcup_{E \in C} \{E(t) = 0\} \right\} \cap \left\{ \bigcap_{E \in C} \{E(t+h) = 1\} \right\} \right\} \right\} = \\ &= \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \bigcup_{F \in X\mathcal{B}'} \left\{ \left\{ \bigcap_{C \in \mathcal{B}'} \{F_C(t) = 0, F_C(t+h) = 1\} \right\} \cap \left\{ \bigcap \{E(t+h) = 1 : E \in (\cup \mathcal{B}' - \{F_C : C \in \mathcal{B}'\})\} \right\} \right\} \right\}, \end{aligned}$$

onde $X\mathcal{B}'$ é o produto cartesiano dos $C \in \mathcal{B}'$, e $F \in X\mathcal{B}'$ é uma função $F : \mathcal{B}' \mapsto \mathcal{E}$ tal que $F(C) = F_C \in C$ para $C \in \mathcal{B}'$, então utilizando os resultados do Lema 3.2.1 e do Corolário 3.2.1, tem-se que:

$$\lim_{h \downarrow 0} \frac{1}{h} \left\{ \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \bigcup_{F \in X\mathcal{B}'} \left\{ \left\{ \bigcap_{C \in \mathcal{B}'} \{F_C(t) = 0, F_C(t+h) = 1\} \right\} \cap \right\} \right\} \right\},$$

$$\left\{ \left\{ \left\{ \left\{ E(t+h) = 1 : E \in (\cup \mathcal{B}' - \{F_C : C \in \mathcal{B}'\}) \right\} \right\} \right\} \right\} =$$

$$\sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} \lim_{h \downarrow 0} \frac{1}{h} \left\{ P \left\{ \bigcup_{F \in \mathcal{X}_{\mathcal{B}'}} \left\{ \bigcap_{C \in \mathcal{B}'} \{F_C(t) = 0, F_C(t+h) = 1\} \right\} \cap \right.$$

$$\left. \left\{ \left\{ \left\{ E(t+h) = 1 : E \in (\cup \mathcal{B}' - \{F_C : C \in \mathcal{B}'\}) \right\} \right\} \right\} \right\} = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} Z_{\mathcal{B}'},$$

onde

$$Z_{\mathcal{B}'} = \begin{cases} \sum_{E \in \mathcal{Z}} w_E(t) \prod_{I \in \cup \mathcal{B}' - \{E\}} U_I(t) & \text{se } \cap \mathcal{B}' \neq \emptyset, \text{ onde} \\ & \mathcal{Z} = \{E \in \cap \mathcal{B}' : (\cup \mathcal{B}' - \{E\}) \cap \overline{(\cup \mathcal{B}' - \{E\})} = \emptyset\}, \\ 0 & \text{caso contrário.} \end{cases}$$

Analogamente,

$$B = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \left\{ \bigcap_{C \in \mathcal{B}'} \left\{ \prod_{E \in C} E(t) = 0, \prod_{E \in C} E(t+h) = 1 \right\} \right\} \cap \right. \\ \left. \left\{ \bigcup_{D \in (\mathcal{B} - \mathcal{B}')} \left\{ \prod_{E \in D} E(t) = 1 \right\} \right\} \right\} =$$

$$\sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \left\{ \bigcup_{F \in \mathcal{X}_{\mathcal{B}'}} \left\{ \right. \right.$$

$$\left. \left\{ \bigcap_{C \in \mathcal{B}'} \{F_C(t) = 0, F_C(t+h) = 1\} \right\} \cap \left\{ \left\{ \left\{ E(t+h) = 1 : E \in (\cup \mathcal{B}' - \{F_C : C \in \mathcal{B}'\}) \right\} \right\} \right\} \right\} \right\}$$

$$\cap \left\{ \bigcup_{D \in (\mathcal{B} - \mathcal{B}')} \left\{ \bigcap_{E \in D} E(t) = 1 \right\} \right\} \right\} =$$

$$\sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \left\{ \bigcup_{F \in X_{\mathcal{B}'}, D \in (\mathcal{B} - \mathcal{B}')} \right\} \right\}$$

$$\left\{ \bigcap_{C \in \mathcal{B}'} \{F_C(t) = 0, F_C(t+h) = 1\} \right\} \cap \left\{ \bigcap \{E(t+h) = 1 : E \in (\cup \mathcal{B}' - \{F_C : C \in \mathcal{B}'\})\} \right\} \right\}$$

$$\cap \left\{ \bigcap_{E \in D} \{E(t) = 1\} \right\} \right\}.$$

Da mesma forma que no cálculo de A, utilizando os resultados do Lema 3.2.1 e do Corolário 3.2.1, tem-se

$$\lim_{h \downarrow 0} \frac{1}{h} \left\{ \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P \left\{ \left\{ \bigcup_{F \in X_{\mathcal{B}'}, D \in (\mathcal{B} - \mathcal{B}')} \right\} \right\} \right\}$$

$$\left\{ \bigcap_{C \in \mathcal{B}'} \{F_C(t) = 0, F_C(t+h) = 1\} \right\} \cap \left\{ \bigcap \{E(t+h) = 1 : E \in (\cup \mathcal{B}' - \{F_C : C \in \mathcal{B}'\})\} \right\} \right\}$$

$$\cap \left\{ \bigcap_{E \in D} \{E(t) = 1\} \right\} \right\} =$$

$$\sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} \lim_{h \downarrow 0} \frac{1}{h} \left\{ P \left\{ \left\{ \bigcup_{F \in X_{\mathcal{B}'}, D \in (\mathcal{B} - \mathcal{B}')} \right\} \right\} \right\}$$

$$\left\{ \bigcap_{C \in \mathcal{B}'} \{F_C(t) = 0, F_C(t+h) = 1\} \right\} \cap \left\{ \bigcap \{E(t+h) = 1 : E \in (\cup \mathcal{B}' - \{F_C : C \in \mathcal{B}'\})\} \right\} \right\}$$

$$\cap \left\{ \bigcap_{E \in D} \{E(t) = 1\} \right\} \right\} = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} \sum_{q=1}^{|\mathcal{B} - \mathcal{B}'|} (-1)^{q+1} \sum_{\substack{\gamma \subseteq (\mathcal{B} - \mathcal{B}') \\ |\gamma|=q}} Z_{\mathcal{B}', \gamma}$$

onde

$$Z_{\mathcal{B}', \gamma} = \begin{cases} \sum_{E \in \mathcal{Z}'} w_E(t) \prod_{I \in (\cup \mathcal{B}' - \{E\})} U_I(t) \prod_{L \in \cup \gamma} U_L(t) & \text{se } \cap \mathcal{B}' \neq \emptyset, \text{ onde} \\ & \mathcal{Z}' = \{E \in \cap \mathcal{B}' : \alpha \cap \bar{\alpha} = \emptyset, \\ & \beta \cap \bar{\beta} = \emptyset, \alpha \cap \bar{\beta} = \emptyset, \\ & \alpha, \beta = \cup \mathcal{B}' - \{E\}, \cup \gamma\}, \\ 0 & \text{caso contrário.} \end{cases}$$

2. A obtenção do número médio de remoções do evento topo, $v_T^*(t)$, é feita de forma análoga à de $w_T^*(t)$. Especificamente,

$$v_T^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{T(t) = 1, T(t+h) = 0\},$$

que pode ser calculada através das expressões derivadas em 1., com as seguintes trocas:

- (1) onde aparecer $E(t) = 0$ trocar por $E(t+h) = 0$
- (2) onde aparecer $E(t+h) = 1$ trocar por $E(t) = 1$
- (3) onde aparecer $w_E(t)$ trocar por $v_E(t)$.

□

Indisponibilidade do Evento Topo T no instante t

Baseados fundamentalmente nas representações para $T(t)$ dadas em (3.2.6) a (3.2.9), vários métodos foram desenvolvidos para o cálculo exato e aproximado da indisponibilidade do evento topo T no instante t , definida por

$$U^*(t) = P\{T(t) = 1\}, \quad t \geq 0.$$

Os métodos para o cálculo exato são aplicáveis quando as estruturas das árvores de eventos não são muito complexas. Porém, quando o evento topo representa uma combinação

complexa de eventos terminais básicos, torna-se difícil, e as vezes até impraticável, o cálculo exato de sua indisponibilidade no instante t . Nestes casos, os métodos aproximados apresentam-se como ferramentas fundamentais.

A seguir são descritos os métodos mais importantes para o cálculo exato e aproximado da indisponibilidade do evento topo T no instante t .

CÁLCULO EXATO

Método da Inclusão-Exclusão

Utilizando uma base \mathcal{B} para T , pode-se calcular $U^*(t)$ através de (3.2.6) como :

$$\begin{aligned} U^*(t) &= P\{T(t) = 1\} \\ &= P\left\{\prod_{C \in \mathcal{B}} \prod_{E \in C} E(t) = 1\right\} \\ &= \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P\left\{\bigcap_{D \in \mathcal{B}'} \left\{\prod_{E \in D} E(t) = 1\right\}\right\} \\ &= \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P\left\{\bigcap_{E \in (\cup \mathcal{B}')} \{E(t) = 1\}\right\}, \quad t \geq 0, \end{aligned}$$

onde

$$P\left\{\bigcap_{E \in (\cup \mathcal{B}')} \{E(t) = 1\}\right\} = \begin{cases} \prod_{E \in (\cup \mathcal{B}')} U_E(t) & \text{se } (\cup \mathcal{B}') \cap (\overline{\cup \mathcal{B}'}) = \emptyset, \\ 0 & \text{caso contrário.} \end{cases}$$

Método da Decomposição de Shannon

Observando que $T(t)$ é uma função binária de $\mathcal{E}_{\mathcal{A}}(t) = \{E(t) : E \in \mathcal{E}_{\mathcal{A}}\}$, isto é,

$$T(t) = f(\mathcal{E}_{\mathcal{A}}(t)), \quad (3.2.10)$$

tem-se que

$$T(t) = E(t)f(1_E, \mathcal{E}_{\mathcal{A}}(t)) + (1 - E(t))f(0_E, \mathcal{E}_{\mathcal{A}}(t)), \quad E \in \mathcal{E}_{\mathcal{A}}, \quad t \geq 0, \quad (3.2.11)$$

onde

$f(1_E, \mathcal{E}_A(t))$ é a expressão resultante de (3.2.10), ao fazer $E(t) = 1$ em $\mathcal{E}_A(t)$

e

$f(0_E, \mathcal{E}_A(t))$ é a expressão resultante de (3.2.10), ao fazer $E(t) = 0$ em $\mathcal{E}_A(t)$.

A expressão (3.2.11) pode ser utilizada para o cálculo exato de $U^*(t)$, uma vez que aplicações repetidas de (3.2.11) resulta em uma expressão da forma

$$T(t) = \sum_{\mathbf{e}} \prod_{i=1}^{|\mathcal{E}_A|} (E(t))^{e_E} (1 - E(t))^{1-e_E} f(\mathbf{e}), \quad t \geq 0,$$

onde a somatória é realizada sobre todos os $2^{|\mathcal{E}_A|}$ conjuntos $\mathbf{e} = \{e_E \in \{0, 1\} : E \in \mathcal{E}_A\}$ e $f(\mathbf{e})$ é o resultado obtido de (3.2.10) substituindo-se \mathcal{E}_A por \mathbf{e} . Dessa forma, pode-se calcular $U^*(t)$ como:

$$\begin{aligned} U^*(t) &= P\{T(t) = 1\} \\ &= \sum_{\mathbf{e}} f(\mathbf{e}) \prod_{i=1}^{|\mathcal{E}_A|} [P\{E(t) = 1\}]^{e_E} [P\{E(t) = 0\}]^{1-e_E} \\ &= \sum_{\mathbf{e}} f(\mathbf{e}) \prod_{i=1}^{|\mathcal{E}_A|} [U_E(t)]^{e_E} [1 - U_E(t)]^{1-e_E}, \quad t \geq 0. \end{aligned}$$

Método da Disjunção Normal Disjunta

Se \mathcal{B} é uma base disjunta para T , então pode-se facilmente calcular $U^*(t)$. Com efeito,

$$\begin{aligned} U^*(t) &= P\{T(t) = 1\} \\ &= P\left\{\bigcup_{C \in \mathcal{B}} \prod_{E \in C} E(t) = 1\right\} \\ &= \sum_{C \in \mathcal{B}} \prod_{E \in C} P\{E(t) = 1\} \\ &= \sum_{C \in \mathcal{B}} \prod_{E \in C} U_E(t), \quad t \geq 0. \end{aligned}$$

Como, em geral, as bases para T não são disjuntas, é importante que se disponha de métodos eficientes para transformar uma base \mathcal{B} qualquer em uma disjunção normal disjunta.

Baseados fundamentalmente na identidade,

$$\bigcup_{i=1}^k G_i = G_1 + G_1^c \cap G_2 + \dots + G_1^c \cap G_2^c \cap \dots \cap G_{k-1}^c \cap G_k, \quad (3.2.12)$$

vários algoritmos foram propostos (cf. [Abr], [AR], [BS], [TV]) para transformar uma base qualquer para um evento topo em uma disjunção normal disjunta.

Para gerar uma disjunção normal disjunta com o menor número de elementos possível, em todos os algoritmos citados, a primeira aplicação da identidade (3.2.12) é feita em $\bigcup_{C \in \mathcal{B}} \cap C$, com os conjuntos suficientes mínimos C_1, \dots, C_k , $k = |\mathcal{B}|$, ordenados de forma que se $r < s$, então $|C_r| < |C_s|$ (Observe que G_i^c aparece em (3.2.12) $k - i$ vezes). A diferença entre esses algoritmos está nos procedimentos estabelecidos para a obtenção de uma representação para cada $M_j = \left[\bigcap_{i=1}^{j-1} G_i^c \right] \cap G_j$, $j = 2, \dots, k$, com $M_1 = G_1$, em uma disjunção normal disjunta. Os termos da disjunção normal disjunta obtida, assim como a sua quantidade, são dependentes destes procedimentos.

Apesar desses métodos terem sido desenvolvidos para o caso de uma base \mathcal{B} onde $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) = \emptyset$, eles são igualmente aplicáveis ao caso onde $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) \neq \emptyset$, adicionando uma simplificação para tratar eventos suficientes mínimos mutuamente exclusivos, evitando assim cálculos desnecessários.

Entre os algoritmos citados, o que melhor explora as possíveis simplificações em M_j , $j = 2, \dots, k$, é o algoritmo proposto por Beichelt & Spross [BS], cuja representação de partida para M_j , $j = 2, \dots, k$, é dada na Proposição 3.2.4 abaixo.

Proposição 3.2.4 (Representação de partida para M_j , $j = 2, \dots, k$)

Seja $\mathcal{B} = \{C_1, \dots, C_k\}$, a base completa para T . Se $T = \bigcup_{i=1}^k (\cap C_i)$, tal que $|C_i| \leq |C_j|$ se $i < j$, então

$$T = \sum_{j=1}^k M_j,$$

onde

$$\begin{aligned} M_1 &= \cap C_1, \\ M_j &= \left[\bigcap_{i \in R_j} \overline{(C_i - C_j)} \right] \cap (\cap C_j), \end{aligned} \quad (3.2.13)$$

$$R_j = \{i < j : \overline{C_i} \cap C_j = \emptyset \text{ e } (\overline{C_i - C_j}) \not\supseteq (\overline{C_s - C_j}) \text{ para todo } s < j, s \neq i\},$$

$j = 2, \dots, k$.

Prova - Como $(\bigcup \overline{C_i}) \cap (\bigcap C_j) = \bigcap C_j$, se $\overline{C_i} \cap C_j \neq \emptyset$, para algum $i < j$, então

$$\begin{aligned} M_j &= \left[\bigcap_{\{i < j : \overline{C_i} \cap C_j = \emptyset\}} (\bigcup \overline{C_i}) \right] \cap (\bigcap C_j) \\ &= \left[\bigcap_{\{i < j : \overline{C_i} \cap C_j = \emptyset\}} \left((\bigcup (\overline{C_i - C_j})) \cup (\bigcup (\overline{C_i \cap C_j})) \right) \right] \cap (\bigcap C_j) \\ &= \left[\bigcap_{\{i < j : \overline{C_i} \cap C_j = \emptyset\}} \left(((\bigcup (\overline{C_i - C_j})) \cap (\bigcap C_j)) \cup ((\bigcup (\overline{C_i \cap C_j})) \cap (\bigcap C_j)) \right) \right] \\ &= \left[\bigcap_{\{i < j : \overline{C_i} \cap C_j = \emptyset\}} \left((\bigcup (\overline{C_i - C_j})) \cap (\bigcap C_j) \right) \right] \\ &= \left[\bigcap_{i \in R_j} \bigcup (\overline{C_i - C_j}) \right] \cap (\bigcap C_j), \end{aligned}$$

onde

$$R_j = \{i < j : \overline{C_i} \cap C_j = \emptyset \text{ e } (\overline{C_i - C_j}) \not\supseteq (\overline{C_s - C_j}) \text{ para todo } s < j, s \neq i\},$$

pois se $(\overline{C_r - C_j}) \supseteq (\overline{C_s - C_j})$, então

$$(\bigcup (\overline{C_r - C_j})) \cap (\bigcup (\overline{C_s - C_j})) = (\bigcup (\overline{C_s - C_j})).$$

□

A transformação de $M_j, j = 2, \dots, k$, dado em (3.2.13), em uma disjunção normal disjunta é feita em $|R_j|$ etapas, onde a r -ésima etapa do procedimento, $1 \leq r \leq |R_j|$, é realizada transformando

$$D_{r-1,j} \cap (\bigcup (\overline{C_{i_r} - C_j})) = \sum_u (\bigcap c_{r-1,j}(u)) \cap (\bigcup (\overline{C_{i_r} - C_j})),$$

em uma disjunção normal disjunta $D_{r,j} = \sum_u (\bigcap c_{r,j}(u))$, onde $i_r = \min(R_j - \{i_1, \dots, i_{r-1}\})$.

Para tanto, cada termo $(\bigcap c_{r-1,j}(u)) \cap (\bigcup (\overline{C_{i_r} - C_j}))$ é transformado em uma disjunção

normal disjunta, se $(c_{r-1,j}(u)) \cap (\overline{C_{i_r} - C_j}) = \emptyset$ (caso contrário, o termo se reduz a $\cap c_{r-1,j}(u)$). Nestes casos, assim como na Proposição 3.2.4,

$$(\cap c_{r-1,j}(u)) \cap (\cup (\overline{C_{i_r} - C_j})) = (\cup_{\{E: E \in \overline{(C_{i_r} - C_j) - c_{r-1,j}(u)}\}} E) \cap (\cap c_{r-1,j}(u)),$$

e a transformação é feita utilizando-se (3.2.12) sobre

$$\cup_{\{E: E \in \overline{(C_{i_r} - C_j) - c_{r-1,j}(u)}\}} E.$$

Beichelt & Spross, observando que os termos gerados na etapa r só afetarão as etapas seguintes $r + 1, r + 2, \dots, |R_j|$, concluíram que a transformação de M_j em uma disjunção normal disjunta terá menor número de elementos, se a transformação de $(\cap c_{r-1,j}(u)) \cap (\cup (\overline{C_{i_r} - C_j}))$, em uma disjunção normal disjunta, for realizada como:

$$(\cap c_{r-1,j}(u)) \cap (A^c + A \cap B^c + \dots + A \cap B \cap \dots \cap C^c),$$

onde

$$n_E = \sum_{s=r+1}^{|R_j|} \mathbb{1}\{E \in \overline{[(C_{i_r} - C_j) - c_{r-1,j}(u)]} \cap (\overline{C_{i_s} - C_j}) : (c_{r-1,j}(u)) \cap (\overline{C_{i_s} - C_j}) = \emptyset\},$$

$$E \in \overline{(C_{i_r} - C_j) - c_{r-1,j}(u)},$$

$$n_A \geq n_B \geq \dots \geq n_C, \quad A, B, C \in \overline{[(C_{i_r} - C_j) - c_{r-1,j}(u)]}, \quad r = 1, \dots, |R_j|.$$

A idéia de considerar essa ordenação é baseada no fato de que os termos dessa escolha serão disjuntos da maior quantidade possível de termos $\cup (\overline{C_{i_s} - C_j})$, e ao mesmo tempo terão a maior quantidade de eventos terminais básicos comuns com os termos $\cup (\overline{C_{i_r} - C_j})$, $s = r + 1, \dots, |R_j|$, $r = 1, 2, \dots, |R_j|$.

Observações:

- O algoritmo proposto por Beichelt & Spross é baseado no algoritmo de Abraham [Abr] e é denominado pelos autores, de algoritmo de Abraham melhorado. Em [BS], Beichelt & Spross fazem uma comparação entre seu algoritmo e os de Abraham [Abr] e de Aggarwal & Rai [AR], comprovando a eficiência de seu algoritmo.
- Schneeweiss descreve em [Sch] um algoritmo para a obtenção de uma disjunção normal disjunta, a partir da base completa para um evento topo, observando que a

cada aplicação da identidade de Shannon (3.2.11) a expressão para o evento topo é escrita como a soma de duas subexpressões disjuntas. Baseado neste fato, o algoritmo é executado através de aplicações repetidas dessa identidade, complementado por simplificações do tipo dado em (2.3.1) (Capítulo 2).

Método de Page & Perry

Uma alternativa para o cálculo de $U^*(t)$, que não envolve a determinação de uma expressão analítica especial para T , foi desenvolvida por Page & Perry [PP]. Este método, denominado pelos autores de TDPP (“Top-Down” Page Perry), foi implementado em um programa na linguagem PASCAL que é dado em [PP], e é aplicável à árvores de eventos construídas utilizando as portas lógicas “AND”, “OR” e “NOT”, onde portas lógicas “NOT” só aparecem combinadas com eventos terminais básicos. O algoritmo é um método “top-down” que opera diretamente sobre a árvore de eventos e consiste de um procedimento computacional recursivo baseado em uma função de conjuntos. O programa exige como entradas a estrutura da árvore e os valores das indisponibilidades pontuais, $U_E(t)$, $t \geq 0$ e $E \in \mathcal{E}$, de seus eventos terminais básicos, no instante t de interesse. Sua eficiência, em termos de tempo de processamento, depende da complexidade da árvore.

Para calcular $U^*(t)$ através deste procedimento, o programa utiliza a função PROB, que age em dois conjuntos. $\text{PROB}(S1, S2)$ representa a probabilidade de todos os eventos no conjunto $S2$ ocorrerem simultaneamente e de pelo menos um evento em $S1$ ocorrer. A primeira chamada da função PROB é realizada com o conjunto $S1$ vazio e o conjunto $S2$ tendo como único elemento o evento T .

Sempre que for realizada a avaliação de uma união de eventos, o método da inclusão-exclusão é utilizado, através das identidades:

$$P\{A \cup B\} = P\{A\} + P\{B\} - P\{A \cap B\}$$

ou

$$\begin{aligned} P\{(A \cup B) \cap C\} &= P\{A \cap C\} + P\{B \cap C\} - P\{A \cap B \cap C\} \\ &= P\{A\} [P\{C\} - P\{B \cap C\}] + P\{B \cap C\}, \end{aligned}$$

se A for independente de B e C , onde A, B e $C \in \sigma'(\mathcal{E})$.

O fluxograma para o cálculo de $U^*(t)$ através deste método é dado na Figura 3.2.2 a seguir.

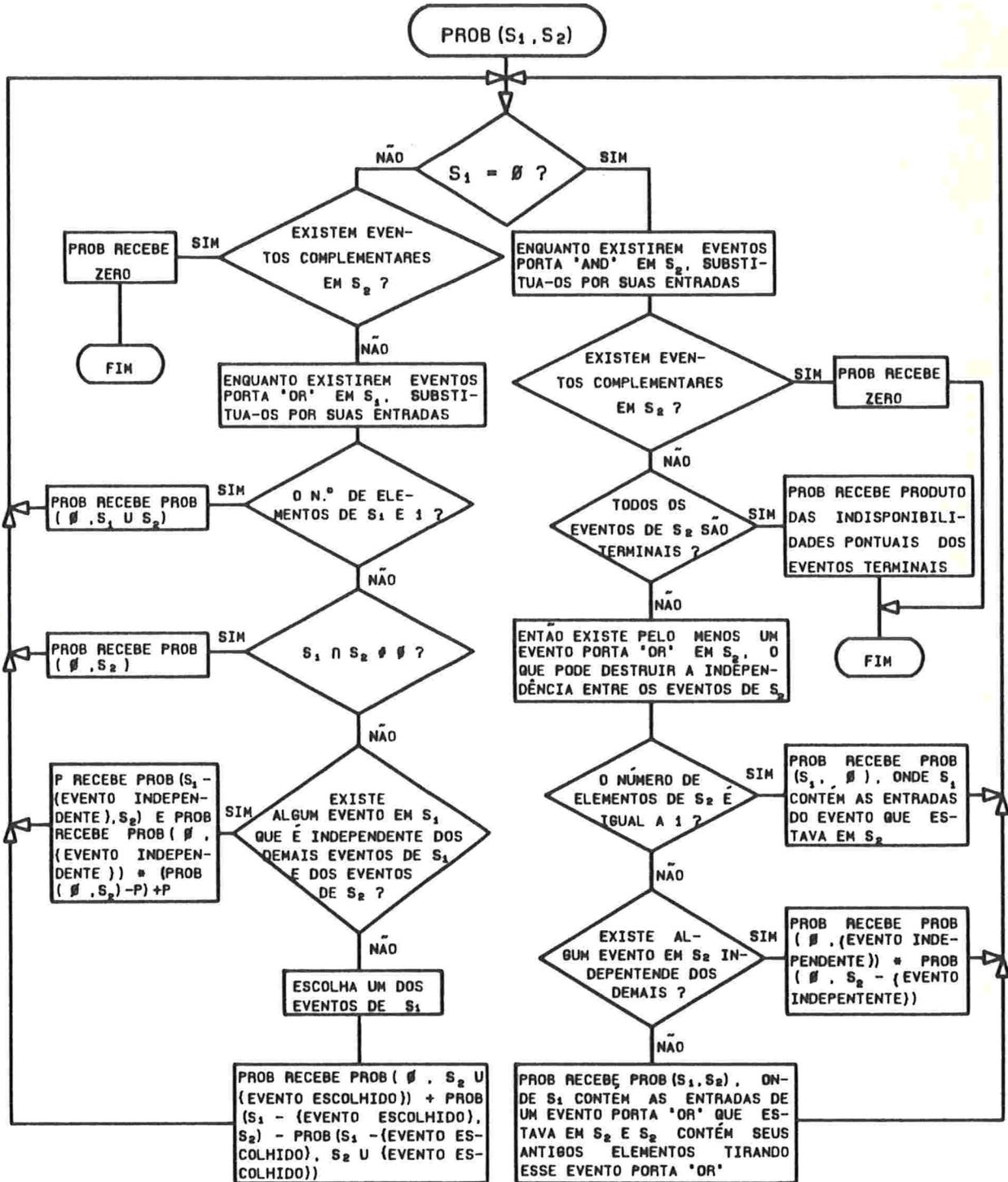


Figura 3.2.2: Fluxograma do Programa TDPP

CÁLCULO APROXIMADO

No caso onde $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) = \emptyset$, Barlow & Proschan [BP] descrevem como utilizar o método da inclusão-exclusão, o método min-max e o método dos eventos cortes/caminhos mínimos, para a obtenção de cotas superiores e inferiores para $U^*(t)$.

Chu & Apostolakis [CA] fizeram um estudo posterior para verificar a aplicabilidade desses métodos no caso onde $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) \neq \emptyset$. Deste estudo, concluiu-se que tanto o método da inclusão-exclusão como o método min-max são igualmente aplicáveis, e que o método dos eventos cortes/caminhos mínimos, que neste caso seriam dados pelos eventos suficientes/necessários mínimos, não é aplicável.

Obtenção de Cotas pelo Método da Inclusão-Exclusão

Pelo método da inclusão-exclusão, tem-se que

$$U^*(t) = \sum_{r=1}^{|\mathcal{B}|} (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P\left\{ \bigcap_{E \in (\cup \mathcal{B}')} \{E(t) = 1\} \right\}, \quad t \geq 0.$$

Assim, pode-se obter cotas para $U^*(t)$ por este método, utilizando as desigualdades de Bonferroni (James [Jam]). Especificamente,

$$U^*(t) \leq \sum_{r=1}^s (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P\left\{ \bigcap_{E \in (\cup \mathcal{B}')} \{E(t) = 1\} \right\}, \quad t \geq 0,$$

se $s \leq |\mathcal{B}|$ e s é ímpar; e

$$U^*(t) \geq \sum_{r=1}^s (-1)^{r+1} \sum_{\substack{\mathcal{B}' \subseteq \mathcal{B} \\ |\mathcal{B}'|=r}} P\left\{ \bigcap_{E \in (\cup \mathcal{B}')} \{E(t) = 1\} \right\}, \quad t \geq 0,$$

se $s \leq |\mathcal{B}|$ e s é par,

onde

$$P\left\{ \bigcap_{E \in (\cup \mathcal{B}')} \{E(t) = 1\} \right\} = \begin{cases} \prod_{E \in (\cup \mathcal{B}')} U_E(t) & \text{se } (\cup \mathcal{B}') \cap (\overline{\cup \mathcal{B}'}) = \emptyset, \\ 0 & \text{caso contrário.} \end{cases}$$

Observações:

- Se $P\{E(t) = 1\} < 0.1$, para todo $E \in (\cup \mathcal{B})$, e o número de conjuntos suficientes mínimos envolvidos não for muito elevado, pode-se obter uma boa aproximação para $U^*(t)$ fazendo $s = 1$. Este fato é conhecido como a lei dos eventos raros. Porém, em situações onde esta condição não é satisfeita, é necessário calcular também cotas para $s = 2, 3, \dots$, para avaliar a qualidade da aproximação.
- Quando $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) \neq \emptyset$, a condição $P\{E(t) = 1\} < 0.1$, para todo $E \in \cup \mathcal{B}$, obviamente não é razoável, pois neste caso, existe pelo menos um par $E, E^c \in \cup \mathcal{B}$.
- Como para $s = 1$ a cota superior fornecida por este método é dada por

$$\sum_{C \in \mathcal{B}} \prod_{E \in C} U_E(t),$$

a melhor aproximação para $U^*(t)$ é obtida, representando $T(t)$ através de uma base irredundante mínima.

Até aqui trabalhou-se com a suposição de independência entre os eventos terminais básicos de $\mathcal{E}_{\mathcal{A}}$. Porém, em certas situações práticas, esta suposição não é razoável. Situações deste tipo surgem, por exemplo, quando componentes no sistema em estudo estão sujeitos ao mesmo conjunto de stresses, ou ainda no caso de componentes que dividem a mesma carga, tal que a falha de um componente resulta em aumento de carga para cada um dos componentes remanescentes. Nestes casos, os eventos terminais básicos são claramente associados.

Em geral, nos problemas de análise de confiabilidade, a associação entre os eventos terminais básicos de $\mathcal{E}_{\mathcal{A}}$ é do tipo não-negativa. Para estes casos, pode-se facilmente obter cotas superiores e inferiores para $U^*(t)$, através do método min-max e do método dos eventos cortes/caminhos mínimos, utilizando a noção de variáveis aleatórias associadas, que é dada na seguinte definição:

Definição 3.2.1 *As variáveis aleatórias X_1, \dots, X_n são ditas associadas, se*

$$Cov[\Gamma(\mathcal{X}), \Delta(\mathcal{X})] \geq 0,$$

para quaisquer funções binárias crescentes Γ e Δ , onde $\mathcal{X} = \{X_i : 1 \leq i \leq n\}$.

Variáveis aleatórias associadas possuem as seguintes propriedades (Barlow & Proschan [BP]):

- (P1) O conjunto consistindo de uma única variável aleatória é associado;
- (P2) Funções crescentes de variáveis aleatórias associadas, são variáveis aleatórias associadas;
- (P3) Se dois conjuntos de variáveis aleatórias associadas são independentes entre si, então sua união é um conjunto de variáveis aleatórias associadas;
- (P4) Variáveis aleatórias independentes são associadas;
- (P5) Se \mathcal{X} é um conjunto de variáveis aleatórias binárias associadas, então $\bar{\mathcal{X}} = \{1 - X_i : 1 \leq i \leq n\}$ é também um conjunto de variáveis aleatórias associadas.

Obtenção de Cotas pelo Método Min-Max

Utilizando as representações (3.2.7) e (3.2.9) para $T(t)$, vale o seguinte Teorema:

Teorema 3.2.2

$$\max_{C \in \mathcal{B}} P\{\min_{E \in C} E(t) = 1\} \leq U(t) \leq \min_{D \in \mathcal{R}} P\{\max_{E \in D} E(t) = 1\}.$$

Além disso, se para cada $C \in \mathcal{B}$ e $D \in \mathcal{R}$, $\{E(t) : E \in C\}$ e $\{E(t) : E \in D\}$ forem conjuntos de variáveis aleatórias associadas, então:

$$\max_{C \in \mathcal{B}} \prod_{E \in C} P\{E(t) = 1\} \leq U(t) \leq \min_{D \in \mathcal{R}} \prod_{E \in D} P\{E(t) = 1\}.$$

Prova - ver Barlow & Proschan [BP].

□

Observação:

- A melhor cota pelo método min-max é obtida, quando a base completa e o recobrimento completo são utilizados.

Obtenção de Cotas pelo Método dos Eventos Cortes/Caminhos Mínimos

Utilizando as representações (3.2.6) e (3.2.8) para $T(t)$, vale o seguinte Teorema:

Teorema 3.2.3 *Se $\{E(t) : E \in \cup \mathcal{B}\}$ e $\{E(t) : E \in \cup \mathcal{R}\}$ são conjuntos de variáveis aleatórias associadas, então*

$$\prod_{D \in \mathcal{R}} P\{\prod_{E \in D} E(t) = 1\} \leq U_T^*(t) \leq \prod_{C \in \mathcal{B}} P\{\prod_{E \in C} E(t) = 1\}.$$

Além disso, se $\{E(t) : E \in \cup \mathcal{B}\}$ e $\{E(t) : E \in \cup \mathcal{R}\}$ forem conjuntos de variáveis aleatórias independentes, então:

$$\prod_{D \in \mathcal{R}} \prod_{E \in D} P\{E(t) = 1\} \leq U^*(t) \leq \prod_{C \in \mathcal{B}} \prod_{E \in C} P\{E(t) = 1\}.$$

Prova - ver Barlow & Proschan [BP].

□

Observações:

- Quando, por exemplo, $(\cup \mathcal{B}) \cap (\overline{\cup \mathcal{B}}) \neq \emptyset$, o Teorema 3.2.3 não se aplica, pois neste caso, existem variáveis aleatórias $E(t)$, $I(t)$ para $E, I \in \cup \mathcal{B}$ tais que $E(t) = 1 - I(t)$. Com isto, $\{E(t) : E \in \cup \mathcal{B}\}$ não pode ser um conjunto de variáveis aleatórias associadas. Para ver isto, sejam

$$\Gamma(\{E(t) : E \in \cup \mathcal{B}\}) = E(t)$$

e

$$\Delta(\{E(t) : E \in \cup \mathcal{B}\}) = I(t),$$

funções binárias crescentes. Assim,

$$\text{Cov}[\Gamma(\{E(t) : E \in \cup \mathcal{B}\}), \Delta(\{E(t) : E \in \cup \mathcal{B}\})] =$$

$$\mathbf{E}[E(t)I(t)] - \mathbf{E}[E(t)]\mathbf{E}[I(t)] =$$

$$\mathbf{E}[(1 - I(t))I(t)] - \mathbf{E}[1 - I(t)]\mathbf{E}[I(t)] =$$

$$-\text{Var}I(t) < 0.$$

Logo, $\{E(t) : E \in \cup \mathcal{B}\}$ não é um conjunto de variáveis aleatórias associadas.

- Um recobrimento para o evento topo T da árvore de eventos \mathcal{A} pode ser obtido a partir das duas primeiras observações do final da Seção 2.2 (Capítulo 2). Assim, se \mathcal{B} é uma base para o evento topo T^c de \mathcal{A}^D , então $\mathcal{R} = \{\bar{C} : C \in \mathcal{B}\}$ é um recobrimento para T . Se a base utilizada for completa, o recobrimento obtido será completo; se a base utilizada for irredundante, o recobrimento obtido será irredundante.
- Apesar dos métodos apresentados nesta seção serem igualmente aplicáveis a uma disjunção normal δ qualquer para T , para se ter certeza de que a análise quantitativa da árvore de eventos necessita ser realizada de uma maneira mais geral, utilizando a metodologia desenvolvida por Inagaki & Henley, é necessário obter a base completa \mathcal{B} para o evento topo da árvore de eventos de interesse. Para entender esta observação, suponha, por exemplo, que a aplicação do algoritmo DN (Capítulo 2) forneceu a disjunção normal $\delta = \{\{A\}, \{A^c, B\}\}$ para o evento topo da árvore. δ satisfaz $(\cup \delta) \cap (\cup \bar{\delta}) \neq \emptyset$ e a árvore de eventos necessita de uma metodologia geral para a análise quantitativa. Porém, a base completa para o evento topo T é $\mathcal{B} = \{\{A\}, \{B\}\}$, que satisfaz $(\cup \mathcal{B}) \cap (\cup \bar{\mathcal{B}}) = \emptyset$ e a árvore pode ser analisada quantitativamente, usando a metodologia desenvolvida por Vesely e Kumamoto & Henley.

Obtidos $U_T^*(t)$, $w_T^*(t)$ e $v_T^*(t)$ e observando que os resultados do Corolário 3.2.1 também valem considerando T ao invés de E , pode-se estabelecer as demais características

do comportamento dinâmico de T , onde as interpretações são análogas às realizadas para eventos terminais básicos e eventos suficientes mínimos. Especificamente:

Intensidade condicional de ocorrência de T :

$$\Lambda_T^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{T(t+h) = 1/T(t) = 0\} = \frac{w_T^*(t)}{1 - U_T^*(t)} \quad \text{se } t \geq 0 \text{ e } U_T^*(t) < 1.$$

Intensidade condicional de remoção de T :

$$\Upsilon_T^*(t) = \lim_{h \downarrow 0} \frac{1}{h} P\{T(t+h) = 0/T(t) = 1\} = \frac{v_T^*(t)}{U_T^*(t)} \quad \text{se } t \geq 0 \text{ e } U_T^*(t) > 0.$$

Número médio de ocorrências de T em $[0, t]$:

$$W_T^*(t) = \int_0^t w_T^*(u) du, \quad t \geq 0.$$

Número médio de remoções de T em $[0, t]$:

$$V_T^*(t) = \int_0^t v_T^*(u) du, \quad t \geq 0.$$

Referências Bibliográficas

- [Abr] ABRAHAM, J. A. An improved algorithm for network reliability. *IEEE Transactions on Reliability*, R28 (1):58-61, 1979.
- [ACCS] AMENDOLA, A.; CLAROTTI, C. A.; CONTINI, S.; SPIZZICHINO, F. *Analysis of complete logical structures in system reliability assessment*. Luxembourg, Commission of The European communities, EUR 6886 EN-JRC-ISPRA, 1980.
- [BL] BARLOW, R. E. & LAMBERT H. E. Introduction to fault tree analysis. In:———. *Reliability and Fault Tree Analysis*. Philadelphia, SIAM, 1975. p.7-35.
- [BP] BARLOW, R. E. & PROSCHAN, F. *Statistical Theory of Reliability and Life Testing*. New York, Holt, c1975. (Internatioanl Series in Decision Processes)
- [BS] BEICHELT, F. & SPROSS, L. An improved Abraham-method for generating disjoint sums. *IEEE Transactions on Reliability*, R36 (1):70-74, 1987.
- [Bel] BELL TELEPHONE LABORATORIES SERIES. *Launch control safety study*. New Jersey, Murray Hill, 1961. v.1, section VII. (The Bell Telephone Laboratories)
- [CA] CHU, T. L. & APOSTOLAKIS G. Methods for probabilistic analysis of noncoherent fault trees. *IEEE Transactions on Reliability*, R29 (5):354-360, 1980.
- [CB] CROSETTI, P. A. & BRUCE, R. A. Commercial application of fault tree analysis. In:———. *Proc Reliability and Maintainability Conf. 9*:230-244, 1970.
- [FHM] FUSSELL, J. B.; HENRY, E. B.; MARSHALL, N. H. MOCUS - a computer program to obtain minimal sets from fault trees. In:———. *Aeroject Nuclear Company*. Idaho Falls, ANCR - 1156, 1974.

- [FPB] FUSSELL, J. B.; POWERS G. J.; BENNETTS, R. G. Fault Trees - A state of the Art Discussion. *IEEE Transactions on Reliability*, R23 (1):51-55, 1974.
- [Fus] FUSSELL, J. B. Computer aided fault tree construction for electrical systems. In:———. *Reliability and fault tree analysis*. Philadelphia, SIAM, 1975. p.37-56.
- [Gha] GHAZALA, M. J. Irredundant Disjunctive and Conjunctive Forms of a Boolean Function. *IBM Journal*, vol. 1:171-176, 1957.
- [Jam] JAMES, BARRY R. *Probabilidade : Um Curso em Nível Intermediário*. Rio de Janeiro, IMPA, 1981.
- [JM] JOHNSTON, B. D. & MATTHEWS, R. H. *Noncoherent structure theory: A review and its role in fault tree analysis*. Safety and Reliability Directorate, SRD R245, 1983.
- [Haa1] HAASL, D. F. Advanced concepts in fault tree analysis. *System Safety Symposium*. Seattle, Boeing Company, 1965.
- [Haa2] HAASL, D. F., et al. Fault tree handbook. *Nuclear Regulatory Commission*. Washington, Office of Nuclear Commission, NUREG-0492, 1981.
- [HK] HENLEY, E. J. & KUMAMOTO, H. *Reliability Engineering and Risk Assessment*. Englewood Cliffs, Prentice-Hall, 1981.
- [HR] HOUSE, R. W. & RADO T. A Generalization of Nelson's Algorithm for Obtaining Prime Implicants. *The Journal of Symbolic Logic*, 30 (1):8-12, 1965.
- [HW] HULME, B. L. & WORRELL R. B. A prime implicant algorithm with factoring. *IEEE Transactions on Computers*, C24 (11):1129-1131, 1975.
- [IH] INAGAKI, T. & HENLEY, E. J. Probabilistic evaluation of prime implicants and top events for noncoherent system. *IEEE Transactions on Reliability*, R29 (5):361-367, 1980.
- [KT] KARLIN, S. & TAYLOR, H. M. *A First Course in Stochastic Processes*. New York, Academic Press, 1975.
- [KH] KUMAMOTO, H. & HENLEY E. J. Top-down algorithm for obtaining prime implicant sets of noncoherent fault trees. *IEEE Transactions on Reliability*, R27 (4):242-249, 1978.

- [Lam] LAMBERT, H. E. *Fault trees for decision making in system analysis*. Lawrence, Livermore Laboratory, University of California, UCRL-51829, 1975.
- [LOKH] LOCKS, M. O. Fault trees, prime implicants, and noncoherence. OGUNBIYI, E. I.; KUMAMOTO, H. & HENLEY E. J.; LOCKS, M. O. *IEEE Transactions on Reliability*, R29 (2):130-135, 1980.
- [Men] MENDELSON, ELLIOTT *Álgebra Booleana e Circuitos de Chaveamento*. Mc Graw-Hill do Brasil, 1977.
- [Nel] NELSON, R. J. *Introduction to automata*. New York, John Wiley, 1968.
- [PP] PAGE, L. B. & PERRY J.E. An algorithm for exact fault-tree probabilities without cut sets. *IEEE Transactions on Reliability*, R35 (5):544-558, 1986.
- [PT] POWERS, G. M. & TOMPKINS, F. C. *Computer-aided synthesis of fault trees for complex processing systems*. Nordhoff, NATO-Advanced Study Institute on Generic Techniques in System Reliability Assessment, 1973.
- [Qui1] QUINE, W. V. A way to simplify truth functions. *The American Mathematical Monthly*, 62 (9):627-631, 1955.
- [Qui2] QUINE, W. V. The Problem of Simplifying Truth Functions. *The American Mathematical Monthly*, 59 (10):521-531, 1952.
- [RA] RAI, S. & AGGARWAL, K. K. An efficient method for reliability evaluation of a general network. *IEEE Transactions on Reliability*, R27 (3):206-211, 1978.
- [Reu] REUSCH, B. Generation of prime implicants from subfunctions and a unifying approach to the covering problem. *IEEE Transactions on Computers*, C24 (9):924-930, 1975.
- [Sch] SCHNEEWEISS, W. G. Disjoint boolean products via Shannon's expansion. *IEEE Transactions on Reliability*, R33 (4):329-332, 1984.
- [TV] TIWARI, R. K. & VERMA, M. An algebraic technique for reliability evaluation. *IEEE Transactions on Reliability*, R29 (4):311-313, 1980.
- [Ves] VESELY, W. E. A time-dependent methodology for fault tree evaluation. *Nuclear Engineering and Design* 13:337-360, 1970.

- [VN] VESELY, W. E. & NARUM R. E. *PREP and KIT : computer codes for the automatic evaluation of a fault tree*. Idaho Falls, Idaho Nuclear Corporation, IN-1349, 1970.
- [Wor1] WORRELL, R. B. *Set equation transformation system (SETS)*. Albuquerque, Sandia Laboratories, SLA-73-0028A, 1974.
- [Wor2] WORRELL, R. B. Using the set equation transformation system in fault tree analysis. In:———. *Reliability and Fault Tree Analysis*. Philadelphia, SIAM, 1975. p.165-185.
- [WS] WORRELL R. B. & STACK, D. W. *A SETS user's manual for the fault analyst*. Albuquerque, Sandia Laboratories, SAND-77-2051, 1978.
- [WSH] WORRELL R. B., STACK, D. W. & HULME, B. L. Prime implicants of non-coherent fault trees. *IEEE Transactions on Reliability, R30* (2):98-100, 1981.
- [Vie] VIEIRA NETO, A. S. *Rally - M : Um Sistema Computacioanl para a Análise de Confiabilidade*. São Paulo, 1988. Dissertação (Mestrado) IPEN/CNEN - S.P.