Instituto de Ciências Matemáticas e de Computação

ISSN - 0103-2569

ASPECTOS TEÓRICOS E DE IMPLEMENTAÇÃO DE CRITÉRIOS DE GERAÇÃO DE SEQUÊNCIAS DE TESTE BASEADOS EM MÁQUINAS DE ESTADO FINITO.

KAREN KIOMI NAKAZATO
JOSÉ CARLOS MALDONADO
SANDRA C.P.F. FABBRI
PAULO CESAR MASIERO

Nº 25

RELATÓRIOS TÉCNICOS DO ICMC

São Carlos
Out./1994

SYSNO 0872265
DATA / /
ICMC - SBAB

Aspectos Teóricos e de Implementação de Critérios de Geração de Seqüências de Teste baseados em Máquinas de Estado Finito

Karen Kiomi Nakazato[1]
José Carlos Maldonado[2]
Sandra C. P. F. Fabbri[3]
Paulo Cesar Masiero[2]

1- Mestranda do ICMSC-USP-São Carlos; Profa. do CEETPS-FATEC-Taquaritinga

2-Professores do ICMSC-USP-São Carlos-Cx.P. 668

3-Doutoranda do IFSC-USP-São Carlos; Profa. do Depto de Computação-UFSCar-Cx.P.676

Conteúdo

1. Introdução 1
2. Máquinas de Estado Finito 1
3. Critérios de Geração de Seqüências de Teste 4
    3.1. Objetivos e Características 4
3.2. Critério DS 5
3.3. Critério UIO 11
    3.4. Critério W 16
    3.5. Critério Wp 18
4. Análise Comparativa dos Critérios 21
    4.1. Generalidade 21
    4.2. Capacidade de Detecção de Falha 22
    4.3. Comprimento das Seqüências de Teste 23
5. Conclusão 23
Bibliografia 24

Apêndice A

A1. Árvore de Teste 26
A2. Lista de Caminhos da Árvore de Teste 26
A3. Lista das Seqüências de Teste geradas pelo critério W 27
A4. Geração da Árvore de Teste 28
A5. Busca de um Nó na Árvore de Teste 29
A6. Geração das Seqüências de Teste 32
A7. Concatenação da Árvore de Teste com o Conjunto de Caracterização W 34

Lista de Tabelas

Tabela 1 : Forma Geral de uma Tabela de Transição 3
Tabela 2 : Tabela de Transição da Máquina M1 4
Tabela 3: Seqüências Distinguíveis da Máquina M1 8
Tabela 4 : Tabela-Xd da Máquina M1. 9
Tabela 5 : Seqüências de Sincronização da Máquina M1. 10
Tabela 6 : Tabela de Rótulos da Máquina M1 14
Tabela 7 : Seqüências UIO da Máquina M1 14
Tabela 8 : P-caminhos da Máquina M1 14
Tabela 9 : Passeios de aresta da Máquina M1. 15
Tabela 10 : Aplicação do Conjunto de Caracterização W na Máquina M1 17
Tabela 11 : Conjuntos Wi da Máquina M1. 20
Tabela12 : Propriedades das MEF necessárias aos critérios 21

Lista de Figuras

Figura 1 : Máquina M1 3
Figura 2 : Árvore de Distinção da Máquina M1. 7
Figura 3 : Árvore de Sincronização da Máquina M1. 8
Figura 4 : Diagrama-Xd da Máquina M1. 9
Figura 5 : Diagrama-b da Máquina M1. 10
Figura 6 : Árvore de Teste da Máquina M1. 17

Aspectos Teóricos e de Implementação de Critérios de Geração de Seqüências de Teste baseados em Máquinas de Estado Finito

Sumário

Neste trabalho é apresentada uma síntese dos principais critérios de geração de seqüências de teste baseados em Máquinas de EStado Finito (MEF). Uma análise comparativa desses critérios é também apresentada. Os principais aspectos de implementação e os respectivos algoritmos são tratados em detalhes, com ênfase no critério W.

1. Introdução

O objetivo deste trabalho é apresentar alguns critérios de geração de seqüências de teste como o DS[GONE70], o W[CHOW78], o UIO[SABN88] e o Wp[FUJI91], baseados em modelos de Máquinas de Estado Finito (MEF), além de analisar alguns aspectos comparativos desses critérios como detecção de falhas e generalidade, estabelecendo ainda uma tabela de propriedades necessárias à aplicação de cada um dos critérios.

Este trabalho está organizado em quatro seções. Nesta seção foram apresentados os principais objetivos. Na Seção 2 é apresentada a técnica de modelagem Máquinas de Estado Finito. A Seção 3 apresenta uma descrição dos critérios de geração de seqüências de teste baseados em MEF citados anteriormente. Na Seção 4 é feita uma análise comparativa entre esses critérios e na Seção 5 é apresentada a conclusão.

2. Máquinas de Estado Finito

Definição 1 : Uma **MEF M** é uma classe de sistemas com um alfabeto de entrada $X=\{x_1,x_2,x_3,\ldots,x_p\}$, um alfabeto de saída

$Z=\{z_1, z_2, z_3, \ldots, z_q\}$, um conjunto de estados $S=\{s_1, s_2, s_3, \ldots, s_n\}$ e um par de funções de caracterização f_z e f_s dados por:

i) $z_v = f_z(s_v, x_v)$ e

ii) $s_{v+1} = f_s(s_v, x_v)$

onde x_v, z_v e s_v são, respectivamente, o símbolo de entrada, o símbolo de saída e o estado da MEF M no instante t_v e s_{v+1} é o estado no instante t_{v+1}.

As MEF podem ser representadas pelas **Tabelas de Transição** através das funções de caracterização f_z e f_s, que representam os símbolos de saída e o próximo estado da MEF em relação aos símbolos de entrada. Estas tabelas representam todos os possíveis pares ordenados (x_v, s_v) de M, onde $x_v \in X$ e $s_v \in S$, e é formada por duas subtabelas Z_v e S_{v+1}, que representam f_z e f_s, respectivamente. As duas subtabelas têm a coluna mais à esquerda em comum que representa todos os possíveis estados e os cabeçalhos das colunas que representam todos os possíveis símbolos de entrada. Graficamente, as MEF podem ter seu comportamento representado através de **Grafos Direcionados**, sendo cada estado representado por um vértice e cada transição representada por uma aresta direcionada, onde cada aresta possui um rótulo x_i / z_i, com $f_z = z_i$ e o estado destino da aresta representando f_s (a partir deste ponto sempre que se referir a um Grafo Direcionado de uma MEF usar-se o termo Máquina). Na Tabela 1 apresenta-se a forma geral de uma Tabela de Transição, onde:

i) o elemento da subtabela Z_v que possui linha s_v e coluna x_v representa $f_z = (s_v, x_v)$, isto é, representa o símbolo de saída gerado por M quando é aplicado o símbolo de entrada x_v sobre M no estado s_v.

ii) o elemento da subtabela S_{v+1} que possui linha s_v e coluna x_v representa $f_s = (s_v, x_v)$, isto é, representa o próximo estado de M após a aplicação do símbolo de entrada x_v sobre M no estado s_v.

Tabela 1 : Forma Geral de uma Tabela de Transição.

	Z_v			S_{v+1}		
S_v \ X_v	x_1	...	x_p	x_1	...	x_p
s_1	$f_z(s_v, x_v)$			$f_s(s_v, x_v)$		
s_2						
...						
s_n						

Exemplo 1 : Seja M1 uma MEF representada pela Figura 1 e pela Tabela 2.

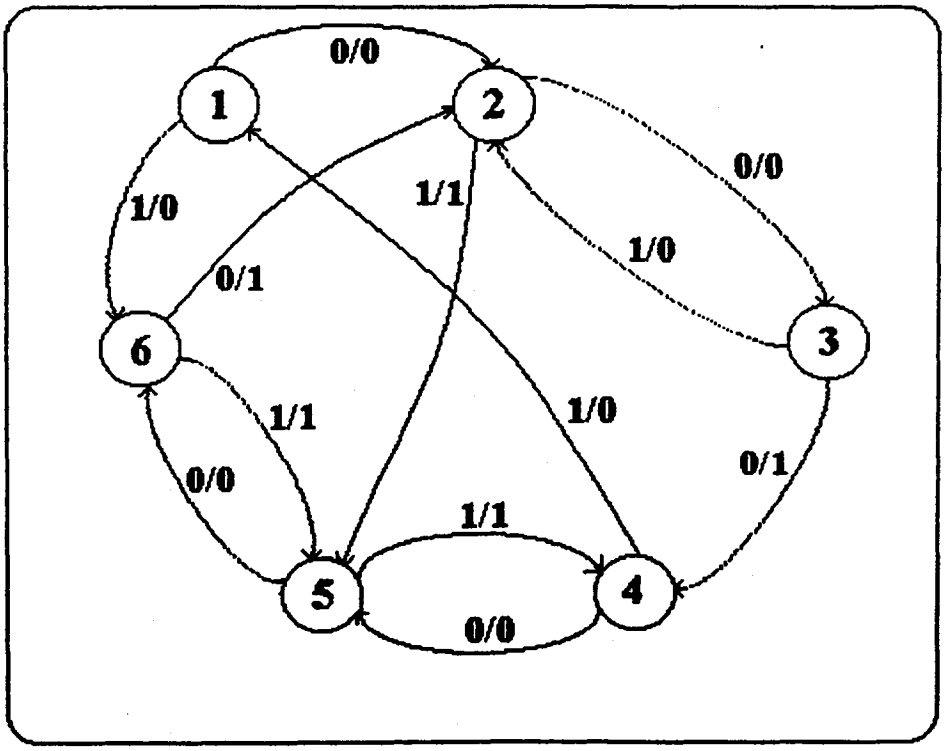


Figura 1 : Máquina M1.

Tabela 2 : Tabela de Transição da Máquina M1.

S\X	0	1	0	1
1	0	0	2	6
2	0	1	3	5
3	1	0	4	2
4	0	0	5	1
5	0	1	6	4
6	1	1	2	5

3. Critérios de Geração de Seqüências de Teste

Além da definição dos objetivos e das principais características dos critérios de geração de seqüências de teste, nesta seção serão apresentadas em detalhes os critérios DS, W, UIO e Wp.

3.1. Objetivos e Características

Um critério de geração de seqüências de teste tem como objetivo, segundo Fujiwara [FUJI91], oferecer a possibilidade de se executar atividades de teste e validação em sistemas modelados de acordo com alguma técnica de modelagem, de forma sistemática, através de procedimentos bem definidos para a geração dessas seqüências, fazendo com que os produtos se apresentem de forma mais segura e com maior qualidade.

Essas seqüências de teste devem, de acordo com Fujiwara[FUJI91], apresentar as seguintes características:

(1) o número pequeno de seqüências , e cada seqüência tendo uma aplicação rápida e de fácil execução.

(2) as seqüências de teste devem identificar todas as falhas que a especificação possa conter.

Os critérios que são apresentados a seguir diferem no tipo de compromisso que é alcançado entre esses objetivos e também na quantidade de formalismo usado para definir cada um desses critérios.

3.2. Critério DS

Este critério, proposto por Gönenç[GONE70], tinha como objetivo inicial a solução do problema de teste de um circuito de chaveamento sequencial, isto é, o problema de determinar se o circuito está ou não operando corretamente de acordo com uma dada Tabela de Transição. Apresentado de forma algorítmica, esse critério é baseado no critério proposto por Hennie[HENN64], mas possui a vantagem de produzir limites no comprimento das seqüências geradas, as quais são então bem menores que as do critério de Hennie.

As MEF através da qual os sistemas são especificados devem apresentar as seguintes propriedades para a aplicação desse critério :

- **Conectividade Forte;**
- **Minimalidade;**
- **Determinismo;**
- **Máquina de Mealy;**
- **Especificação Completa.**

As propriedades necessárias à aplicação desse critério têm suas definições e algoritmos de verificação apresentadas em [NAKA94a].

Este critério exige ainda que as MEF possuam as seguintes seqüências básicas:

- **Seqüência de Sincronização.**
- **Seqüência Distinguível.**

Os algoritmos de geração dessas seqüências são apresentados em [NAKA94b].

O procedimento que define o critério DS sobre uma MEF M pode ser apresentado da seguinte forma :

(1) Geração da Seqüência Distinguível .
(2) Geração do Diagrama-Xd .
(3) Geração das Seqüências de Sincronização .
(4) Geração da Seqüência Inicial .
(5) Geração da Seqüência de Checagem .

No passo (1), com as informações da Tabelas de Transição de M, são geradas todas as Seqüências Distinguíveis de M [NAKA94b] e a mais curta Seqüência Distinguível é denominada Xd.

Para a geração do **Diagrama-Xd** do **passo (2)**, estabelece-se uma tabela, a Tabela Xd, que apresenta para todo estado s_i de M, a seqüência de saída gerada e o estado destino após a aplicação de Xd no estado s_i.

No **passo (3)**, são geradas as Seqüências de Sincronização de M, de acordo com as informações da Tabela de Transição de M e do algoritmo apresentado em [NAKA94b].

Para a geração da Seqüência Inicial do **passo (4)**, deve-se primeiramente identificar no Diagrama-Xd quais os estados que não são destino de nenhuma aresta, os chamados estados origem. Entre os estados origem, o estado que apresentar a Seqüência de Sincronização mais curta será identificado como estado de partida. Assim, a **Seqüência Inicial** será a Seqüência de Sincronização do estado de partida.

No **passo (5)** é executada a geração da **Seqüência de Checagem**. Esta seqüência é formada pela concatenação de duas outras: a a-seqüência e a b-seqüência. O objetivo da **a-seqüência** é reconhecer todos os estados da MEF M e todos os próximos estados de M após da aplicação de Xd sobre cada um dos estados. A **b-seqüência** tem como objetivo verificar cada transição da Tabela de Transição.

A a-seqüência é gerada percorrendo-se o Diagrama-Xd, iniciando num estado origem. Cada estado alcançado pela a-seqüência é marcado e quando um estado s_i é alcançado e já foi marcado anteriormente, é incorporada à a-seqüência uma T-seqüência que leva M do estado s_i a um estado origem, até que todos os estados do Diagrama-Xd tenham sido marcados.

A b-seqüência é gerada tendo como ponto de partida o estado final de M após a aplicação da a-seqüência.

Para a geração da b-seqüência é necessário que seja construído o **Diagrama-b** da seguinte forma : Para todo estado s_i de M que será representado por um vértice do Diagrama-b, devem ser traçadas p arestas que representam as seqüências a_iXd, com $a_i \in X$. Os estados destino dessas arestas são os estados finais de M após a aplicação dessas seqüências.

A geração da b-seqüência é equivalente a determinar a cobertura mínima do Diagrama-b. Quando um grafo que representa o Diagrama-b não é fortemente conectado, a cobertura mínima do Diagrama-b é formada pelas coberturas mínimas de seus componentes. As coberturas

mínimas são representadas por seqüências de estados e hífens que separam cada uma das coberturas mínimas de cada um dos componentes do Diagrama-b.

O algoritmo utilizado para a geração da cobertura mínima do Diagrama-b é baseado no algoritmo de Fleury para a geração de cobertura mínima para grafos não direcionados Eulerianos [BERG62].

A b-seqüência é descrita por uma cobertura mínima do Diagrama-b da seguinte forma:

$$g_{i1j1}g_{i1j2}\cdots g_{i1jm}-g_{i2j1}g_{i2j2}\cdots g_{i2jn}-\cdots g_{itj1}g_{itj2}\cdots g_{itjs}$$

Cada par $g_{ivjw}g_{ivjw+1}$ será representado pela aresta do Diagrama-b, e tem como estado origem g_{ivjw} e como estado destino g_{ivjw+1}. Esta aresta tem como rótulo uma seqüência de símbolos de entrada que será incorporada à b-seqüência. Quando ocorre um **hífen**, é gerada uma **T-seqüência** que leva M do estado à esquerda do hífen para o estado à direita do hífen. Esta T-seqüência também é incorporada à b-seqüência.

Exemplo 1 : Seja M1 a MEF definida na Figura 1 com seu comportamento descrito na Tabela 1, que possui todas as propriedades e seqüências básicas exigidas pelo critério DS. Assim, tem-se na Figura 2 a Árvore de Distinção e na Figura 3 a Árvore de Sincronização de M1.

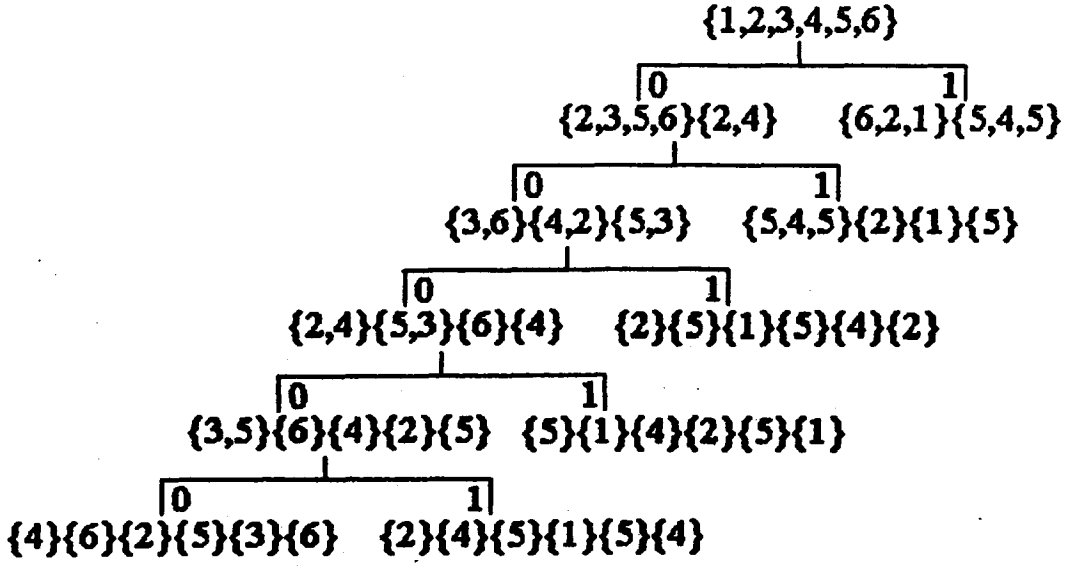


Figura 2 : Árvore de Distinção da Máquina M1.

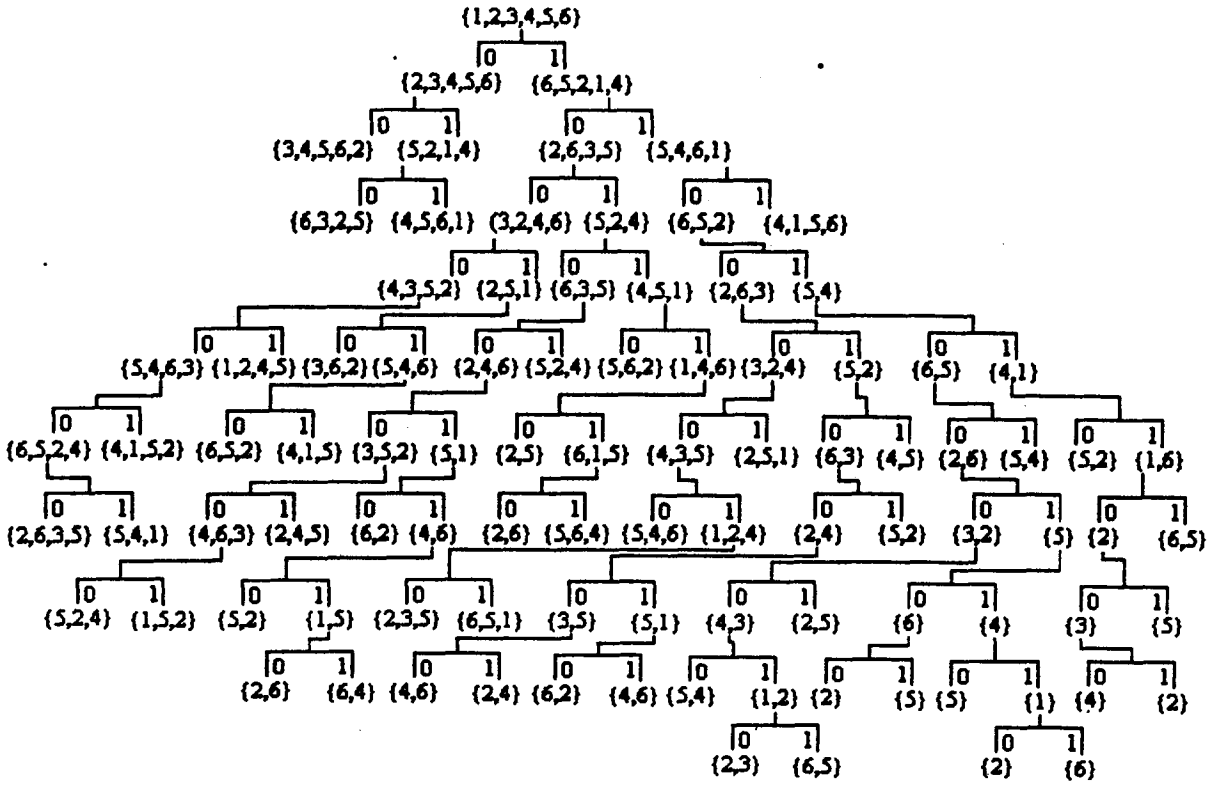


Figura 3 : Árvore de Sincronização da Máquina M1.

De acordo com a Árvore de Distinção da Figura 2, as Seqüências Distinguíveis de M1 são apresentadas na Tabela 2 .

Tabela 3: Seqüências Distinguíveis da Máquina M1.

DS1(A1)	**001**
DS2(A1)	**0001**
DS3(A1)	**00000**
DS4(A1)	**00001**

De acordo com a Tabela 3, a mais curta DS é a DS1 então Xd=001. Assim com as informações da Tabela de Transição de M1 e de Xd, estabelece-se a Tabela-Xd de M1 representada pela Tabela 4.

Tabela 4 : Tabela-Xd da Máquina M1.

estado	próximo estado	seqüência de saída
1	2	000
2	1	010
3	4	101
4	5	001
5	5	011
6	2	100

Em seguida são construídos o Diagrama-Xd da Figura 4 e o Diagrama-b da Figura 5.

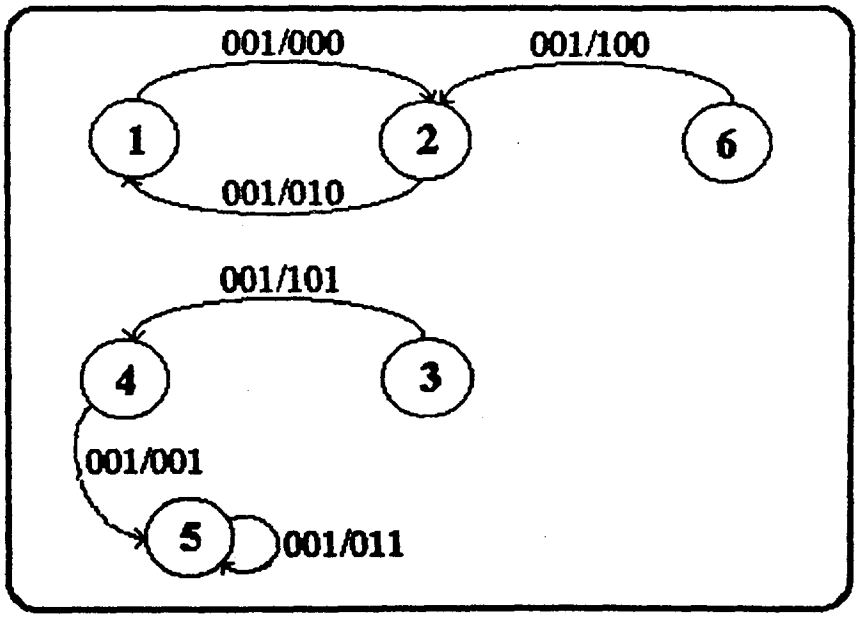


Figura 4 : Diagrama-Xd da Máquina M1.

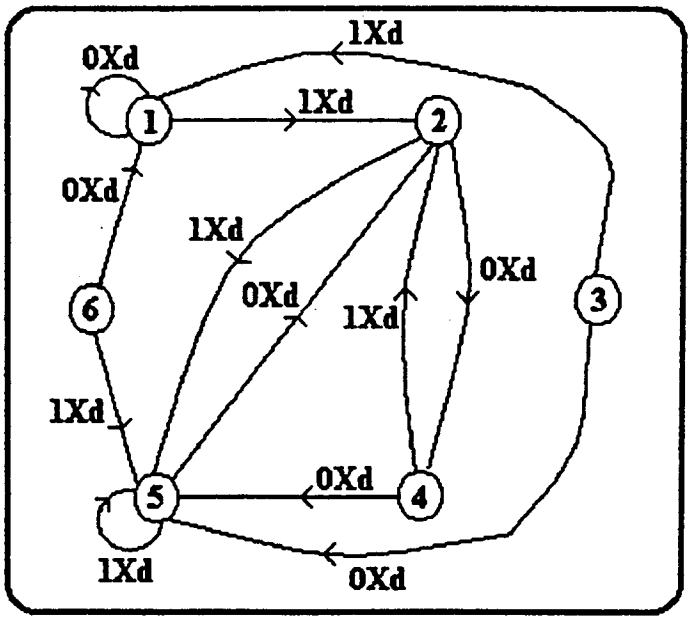


Figura 5 : Diagrama-b da Máquina M1.

De acordo com a Figura 4, pode-se identificar dois estados denominados origem, isto é, estados que não são destino de nenhuma aresta. Assim 3 e 6 são estados origem do Diagrama-Xd .

A Tabela 5 apresenta todas as Seqüências de Sincronização de M1 segundo a Árvore de Sincronização da Figura 3.

Tabela 5 : Seqüências de Sincronização da Máquina M1.

SS(1)	110100111
SS(2)	1101110
SS(3)	11011100
SS(4)	11010011
SS(5)	1101001
SS(6)	11010010

Para a geração da Seqüência Inicial, deve-se selecionar a Seqüência de Sincronização mais curta dos estados origem de Xd; mas, de acordo com a Tabela 5, as Seqüências de Sincronização dos estados 3 e 6 têm o mesmo comprimento; assim, seleciona-se o estado 3, então o estado 3 é denominado estado de partida. A Seqüência Inicial é a Seqüência de

Sincronização do estado de partida , dessa forma tem-se que a seqüência inicial = 11011100.

A geração da Seqüência de Checagem iniciada com a geração do Diagrama-b, é formada pela concatenação de duas outras seqüências , a a-seqüência e a b-seqüência que de acordo com o Diagrama-Xd e o Diagrama-b da MEF M1 e representados pelas Figuras 4 e 5 respectivamente, são descritos da seguinte forma :

a-seqüência =

00100100100100001001001001

b-seqüência =

000110011001100100010001100100001000101001000010001001

A b-seqüência foi gerada através da seguinte cobertura mínima do Diagrama-b da Figura 5 :

11255242-35-45-65-61-31

Assim para a MEF M1, o critério DS gera as seqüências inicial e de checagem.

3.3. Critério UIO

Proposto por Sabnani[SABN88], este critério teve como principal motivação a checagem de conformidade de especificações de protocolos de comunicação de redes de computadores. Para Sabnani uma seqüência de teste é uma seqüência de símbolos de entrada e saída. Assim, a checagem de conformidade consiste em verificar se os símbolos de saída gerados são idênticos aos das saídas esperadas .

Para que o critério UIO possa ser aplicado, as MEF devem apresentar as seguintes propriedades :

- **Conectividade Forte;**
- **Minimalidade;**
- **Determinismo;**
- **Máquina de Mealy;**
- **Especificação Completa;**
- **Estado inicial fixo.**

A menos da última propriedade, todas as outras têm suas definições além dos algoritmos de verificação apresentados em [NAKA94a].

O critério UIO exige ainda que a MEF M apresente as seguintes seqüências básicas:

- Seqüências Únicas de entrada/saída para cada estado de M.

O procedimento que define o critério UIO pode ser apresentado da seguinte forma :

(1) Geração da Tabela de Rótulos.

(2) Geração de seqüências únicas de entrada/saída para cada estado de M.

(3) Geração dos P-caminhos para cada estado.

(4) Geração dos passeios de aresta para cada aresta de M.

(5) Geração do passeio de todas as arestas.

Neste critério os estados são representados pelos vértices do grafo direcionado da MEF M e cada aresta representa uma possível transição de estado de M que possui um rótulo **i/o**, onde **i** é um símbolo de entrada e **o** é um símbolo de saída da máquina.

Para a geração da Tabela de Rótulos são identificados todos os rótulos apresentados por M e para cada rótulo é apresentada uma lista de transições onde cada elemento da lista, $o_i d_j$, representa o vértice origem o_i e o vértice destino d_j de cada aresta que possui este rótulo.

O algoritmo de geração das Seqüências Únicas de entrada/saída para cada estado é apresentado em [NAKA94b].

A geração dos P-caminhos para cada estado de M representa calcular os mais curtos caminhos do estado inicial até cada um desses estados. Neste critério o algoritmo usado para geração dos P-caminhos é o de Djkstra [AHO 76].

A geração dos passeios de aresta para cada aresta da MEF M é feita através das seguintes sub-seqüências:

TE(aresta) = P((Head(aresta))@rótulo(aresta)@UIO(Tail(aresta))

onde:

TE(aresta) é o passeio da aresta

Head(aresta) é o estado origem da aresta

Tail(aresta) é o estado final da aresta

Dessa forma, cada passeio de aresta verifica se cada aresta tem o estado origem, o estado destino e o rótulo corretos.

A geração do passeio de todas as arestas é feito da seguinte forma :

TS=#ri/null@TE(aresta)

onde ri/null é uma operação que leva a MEF M para o estado inicial e não produz nenhum símbolo de saída e # é a concatenação de todas as sub-seqüências ri/null@TE(aresta), para toda aresta de M.

Exemplo 2 :Seja M1 a MEF definida na Figura 1, com sua Tabela de Transição na Tabela 1 e que apresenta todas as propriedades necessárias à aplicação do critério UIO. Então de acordo com a Tabela 1 pode-se estabelecer a Tabela de Rótulos de M1 na Tabela 6 e a Tabela de Seqüências Únicas de entrada/saída na Tabela 7, além dos P-caminhos na Tabela 8.

Tabela 6 : Tabela de Rótulos da Máquina M1.

Rótulo	Lista de Transições
0/0	12,23,45,56
0/1	62,34
1/0	16,41,32
1/1	65,25,54

Tabela 7 : Seqüências UIO da Máquina M1.

UIO(1)	1\0.0\1
UIO(2)	0\0.1\0
UIO(3)	0\1.1\0
UIO(4)	1\0.1\0
UIO(5)	1\1.1\0
UIO(6)	0\1.1\1

Tabela 8 : P-caminhos da Máquina M1.

P(1)	{ }
P(2)	0/0
P(3)	0/0.0/0
P(4)	0/0.0/0.0/1
P(5)	0/0.1/1
P(6)	1/0

Os passeios de aresta para cada aresta de M1 são apresentados na Tabela 9.

Tabela 9 : Passeios de aresta da Máquina M1.

TE(16)	1/0.0/1.1/1
TE(41)	0/0.0/0.0/1.1/0.1/0.0/1
TE(32)	0/0.0/0.1/0.0/0.1/0
TE(65)	1/0.1/1.1/1.1/0
TE(54)	0/0.1/1.1/1.1/0.1/0
TE(12)	0/0.0/0.1/0
TE(23)	0/0.0/0.0/1.1/0
TE(45)	0/0.0/0.0/1.0/0.1/1.1
TE(56)	0/0.1/1.0/0.0/1.1/1
TE(62)	1/0.0/1.0/0.1/0
TE(34)	0/0.0/0.0/1.1/0.1/0
TE(25)	0/0.1/1.1/1.1/0

Finalmente apresenta-se o passeio de todas as arestas denominado TS que representa a seqüência de teste gerada pelo critério UIO.

TS = [ri/null.1/0.0/1.1/1]
[ri/null.0/0.0/0.0/1.1/0.1/0.0/1]
[ri/null.0/0.0/0.1/0.0/0.1/0]
[ri/null.1/0.1/1.1/1.1/0]
[ri/null.0/0.1/1.1/1.1/0]
[ri/null.0/0.1/1.1/1.1/0.1/0]
[ri/null.0/0.0/0.1/0]
[ri/null.0/0.0/0.0/1.1/0]
[ri/null.0/0.0/0.0/1.0/0.1/1.1/0]
[ri/null.0/0.1/1.0/0.0/1.1/1]
[ri/null.1/0.0/1.0/0.1/0]
[ri/null.0/0.0/0.0/1.1/0.1/0]

onde ri é uma função que leva a MEF M1 de qualquer estado para o estado inicial e que produz uma saída nula **null**.

A seguir apresenta-se o critério W, além de um exemplo de aplicação do mesmo usando a MEF M1 dos exemplos anteriores .

3.4. Critério W

Proposto por Chow[CHOW78], este critério de geração de seqüências de teste baseado na modelagem de projetos de sistema através de Máquinas de Estado Finito tem as seguites características :

(1) Somente a estrutura de controle do projeto é checada .

(2) Não existe a exigência de especificação executável .

Os sistemas considerados neste critério apresentam dois tipos de primitivas : estímulos e operações; estímulos são entradas do mundo real para o sistema e operações são eventos causados pelo sistema pela ativação de operações como resposta a algum estímulo. A existência de uma grande classe de sistemas nas áreas de Análise Léxica, Processamento de Dados, Controle de Processos em Tempo Real e outros que apresentam as características dos sistemas considerados no critério foram uma das principais motivações .

As MEF que podem ser submetidas à aplicação deste critério devem apresentar as seguintes propriedades :

- **Minimalidade;**
- **Determinismo;**
- **Máquina de Mealy;**
- **Especificação Completa;**
- **Estado inicial fixo;**
- **Todo estado ser alcançável a partir do estado inicial.**

As quatro primeiras propriedades têm sua definição e algoritmos de identificação apresentados em [NAKA94a] e [NAKA94b].

Para se verificar a alçancabilidade de todo estado de M a partir do estado inicial é usado o algoritmo de Dyjkstra[Aho76].

A MEF M deve ainda apresentar:

- **O Conjunto de Caracterização W.**

O critério W pode ser descrito através do seguinte procedimento :

(1) Geração do Conjunto de Caracterização W.

(2) Geração da Árvore de Teste.

(3) Geração das seqüências de teste.

A geração do Conjunto de Caracterização têm seu algoritmo apresentado em [NAKA94b].

A geração da Árvore de Teste e da seqüência de teste são apresentados no Apêndice A.

Exemplo 3 : Seja M1 a MEF da Figura 1 e com a Tabela de Transição na Tabela 1, apresentando todas as propriedades necessárias à aplicação do critério W .

É gerado o Conjunto de Caracterização de M1; assim, de acordo com algoritmo de geração de W de [NAKA94b], W = {{0},{1},{11},{10}} .

Pode-se estabelecer a Tabela 10 que ilustra o comportamento que define o Conjunto de Caracterização de M1, em relação a todos os seus estados e a Árvore de Teste de M1 na Figura 6.

Tabela 10 : Aplicação do Conjunto de Caracterização W na Máquina M1 .

Sv \ W	0	1	11	10
1	0	0	01	01
2	0	1	11	10
3	1	0	01	00
4	0	0	00	00
5	0	1	10	10
6	1	1	11	10

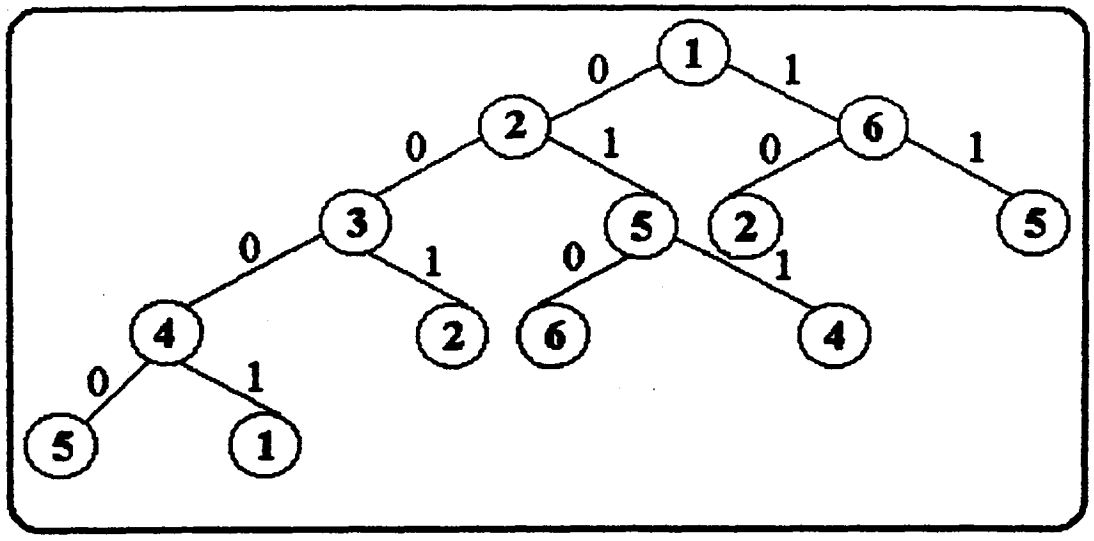


Figura 6 : Árvore de Teste da Máquina M1.

Finalmente são geradas as seqüências de teste da MEF M1 :

00,01,010,011
10,11,110,111
000,001,0010,0011
010,011,0110,0111
100,101,1010,1011
110,111,1110,1111
0000,0001,00010,00011
0010,0011,00110,00111
0100,0101,01010,01011
0110,0111,01110,01111
00000,00001,000010,000011
00010,00011,000111.000110
01000,01001,010010,010011
01010,01011,010110,010111

Para aplicação dessas seqüências sobre a MEF M1 é necessário que para cada seqüência aplicada em M1 retorne-se ao estado inicial.

A seguir, é apresentado o critério Wp, juntamente com um exemplo de aplicação também sobre a MEF M1 .

3.5. Critério Wp

Proposto por Fujiwara[FUJI91], este critério tem como uma das principais motivações o fato de que métodos para o desenvolvimento de seqüências de teste têm recebido muita atenção em relação a testes de conformidade de protocolos de comunicação de redes de computadores [RAYN87] [SARI89] .

Como este critério também é baseado na modelagem de sistemas através de Máquinas de Estado Finito, é necessário que uma MEF M apresente as seguintes características :

- **Minimalidade;**
- **Determinismo ;**
- **Máquina de Mealy;**
- **Especificação Completa;**
- **Estado inicial fixo;**

- Todo estado ser alcançável a partir do estado inicial.

As quatro primeiras propriedades têm sua definição e algoritmos de identificação apresentados em [NAKA94a] e [NAKA94b].

Para se verificar a alçancabilidade de todo estado a partir do estado inicial é usado o algoritmo de Dyjkstra[Aho76].

A MEF M deve ainda apresentar:

- O Conjunto de Caracterização W.

Dessa forma essas propriedades são as mesmas exigidas pelo critério W.

O procedimento que define este critério sobre uma MEF M que possui as propriedades citadas anteriormente pode ser apresentado da seguinte forma :

(1) Geração do Conjunto de Caracterização W.

(2) Geração dos Conjuntos Wi.

(3) Geração dos Conjuntos Q, P e R.

(4) Geração das seqüências da Fase 1.

(5) Geração das seqüências da Fase 2.

Os Conjuntos Wi podem ser definidos como Conjuntos de Caracterização de cada estado S_i de M; dessa forma, estes conjuntos identificam unicamente cada estado de M e são subconjuntos do Conjunto de Caracterização W.

O Conjunto Q é o conjunto de seqüências de entrada que levam a MEF M do estado inicial para todos os outros estados e, por definição, tem-se que a seqüência nula, que leva M do estado inicial para o próprio estado inicial, também está incluída em Q.

O Conjunto P é o conjunto que representa todos os caminhos parciais da Árvore de Teste do critério W.

O Conjunto R é formado pelas seqüências que estão no Conjunto P e não se encontram no Conjunto Q.

A Fase 1 tem como objetivo verificar se todos os estados definidos pela especificação são identificáveis na implementação e ao mesmo tempo verificar se as transições que levam M do estado inicial para esses estados apresentam as saídas e o próximo estado corretos.

A Fase 2 verifica todas as transições definidas pela Tabela de Transição, mas que não foram checadas na Fase 1.

Apresenta-se a seguir um exemplo de aplicação do critério Wp sobre a MEF M1, uma vez que M1 possui todas as propriedades necessárias ao critério.

Exemplo 4 : Seja M1 a MEF da Figura 1 com sua Tabela de Transição na Tabela 1. O Conjunto de Caracterização W da Tabela 10 pode ser reutilizado e os conjuntos Wi, que têm seu algoritmo de geração especificado em [FUJI91], podem ser estabelecidos na Tabela 11.

Tabela 11 : Conjuntos Wi da Máquina M1.

W1	{0,11}
W2	{1,10,11}
W3	{0,10}
W4	{1,11}
W5	{1,0,11}
W6	{0,1}

De acordo com os algoritmos de geração de Q, P e R de [FUJI91], tem-se para a MEF M1 :

Q = {0,00,000,11,1}

P = {0,1,00,01,10,11,000,001,010,011,0000,0001,0100,0101}

R = {01,10,001,010,011,0000,0001,0100,0101}

As seqüências da Fase 1 são denotadas pelo Conjunto Q.W que também tem seu algoritmo especificado em [FUJI91], assim em relação a MEF M1 tem-se :

Q.W = {0,1,10,11,00,01,010,011,000,001,
0010,0011,0000,0001,00010,00011,
110,111,1110,1111 }

O conjunto ROW que representa as seqüências da Fase 2 é gerado por um algoritmo apresentado também em [FUJI91], assim para a MEF M1 tem-se :

ROW = {010,1001,1011,00101,00111,0100,01010,
0111,01111,00000,000011,00010,000111,
01001,010001,010011,01011,01010,010111}

A seguir são analisados em relação a cada um dos critérios apresentados anteriormente, alguns aspectos que podem, de alguma forma, justificar a aplicação de um determinado critério sobre uma MEF.

4. Análise Comparativa dos Critérios

Alguns aspectos tornam-se bastante importantes para a seleção de um determinado critério de geração de seqüências de teste como a identificação de algumas propriedades sobre as MEF, a aplicabilidade e o comprimento das seqüências geradas entre outros, que são abordados a seguir em relação aos critérios já apresentados anteriormente neste trabalho.

4.1. Generalidade

Para uma análise comparativa entre as propriedades necessárias a cada um dos critérios, é estabelecida a Tabela 12 .

Tabela12 : Propriedades das MEF necessárias aos critérios .

Propriedade \ Critério	W	Wp	UIO	DS
Minimalidade	X	X	X	X
Especificação Completa	X	X	X	X
Conectividade Forte	—	—	X	X
Máquina de Mealy	X	X	X	X
Determinismo	X	X	X	X
Alcançabilidade do estado inicial	X	X	—	—
Fixação do estado inicial	X	X	—	—
Seqüência de Sincronização	—	—	—	X
Seqüência Distinguível	—	—	—	X
Seqüência Única de Entrada/Saída	—	—	X	—
Conjunto de Caracterização	X	X	—	—

De acordo com a Tabela 12, os critérios W e Wp exigem minimalidade, especificação completa, determinismo, fixação do estado inicial e que seja uma máquina de Mealy; mas, na aplicação das seqüências de teste é necessário ainda que uma função leve a MEF de qualquer estado para o estado inicial. Entretanto em comparação aos outros critérios apresentados e que não fazem uso dessa funcão, não é necessário que a MEF seja fortemente conectada .

Entre todos os critérios apresentados ,W e Wp são os únicos de aplicabilidade geral no sentido que o Conjunto de Caracterização W e os conjuntos Wi existem para toda máquina minimal. A limitação de aplicabilidade dos critérios DS e UIO se justifica pelo fato de que mesmo as MEF sendo minimais, completamente especificadas e fortemente conectadas, não há garantia de existência de Seqüências Distinguíveis e/ou Seqüências Únicas de entrada/saída .

4.2. Capacidade de Detecção de Falha

Para MEF completamente especificadas, seqüências de teste geradas pelos quatro critérios são capazes de detectar falhas em saídas de todas as transições .

Como essas seqüências são geradas praticamente da mesma forma, exceto pelas diferentes seqüências de identificação [NAKA94b] para reconhecerem os estados, é esperado que tenham a mesma capacidade de detecção de falha. Essa capacidade é definida no sentido que são identificadas falhas nas saídas geradas pelas seqüências de entradas, nos estados destino de cada transição e na checagem das saídas esperadas pela especificação e as geradas pela MEF durante a aplicação das seqüências de teste, além da identificação de **erros de transferência** (próximo estado incorreto), de **operação** (símbolo de entrada ou símbolo de saída da transição incorretos) e de **estados em excesso ou falta** na MEF [CHOW78].

4.3. Comprimento das Seqüências de Teste

De acordo com Sidhu[SIDH89], o principal fato que contribui para o comprimento das seqüências de teste reside na escolha das seqüências de identificação; Dessa forma, se o comprimento de uma Seqüência Distinguível do critério DS é muito maior que o das as Seqüências Únicas de entrada/saída do critério UIO, então o critério DS gerará seqüências muito mais longas que o critério UIO e assim discussões similares podem ser feitas entre os outros critérios . Ainda de acordo com Sidhu[SIDH89], segundo aplicações práticas, o critério UIO produz a seqüência de teste mais curta enquanto que entre os critérios DS, UIO e W, W produz a seqüência mais longa, embora Fujiwara[FUJI91] demonstre que o critério Wp gere seqüências ainda mais curtas que o critério W através do uso dos conjuntos Wi.

5. Conclusão

Foi apresentado neste trabalho um estudo dos critérios DS, UIO, W e Wp de geração de seqüências de teste baseados em Máquinas de Estado Finito, onde os procedimentos definidos para cada critério e os algoritmos que implementam esses procedimentos foram descritos. Complementando-se este estudo, cada um dos critérios foi aplicado numa mesma MEF M1, permitindo que uma análise comparativa em relação às seqüências geradas pelos critérios pudesse ser realizada.

Para cada um dos critérios, a MEF deve satisfazer algumas propriedades e dessa forma o critério W apresentou maior aplicabilidade uma vez que um número bem maior de MEF satisfaz as propriedades exigidas pelo critério W; assim os aspectos de implementação deste critério foram melhor explorados, permitindo porém, que os outros critérios possam ser implementados baseando-se nas informações obtidas neste trabalho. Para a verificação das propriedades e a aplicação de cada um dos critérios, foram utilizados os algoritmos e estruturas de dados de [NAKA94a] e [NAKA94b].

Bibliografia

[AHO76] AHO, A.V., HOPCROFT, J.E. and ULMANN, J.D., The Design and Analysis of Computer Algorithms, Addison-Wesley, Reading, MA, 1976.

[BERG62] BERGE, C. Theory of Graphs, London: Methuen, 1962.

[CHOW78] CHOW, T. S. "Testing Software Design modeled by Finite State Machines" in IEEE Trans. on Software Engineering, vol.SE-4, March 1978, pp 178-187.

[FUJI91] FUJIWARA, S. "Test Selection Based on Finite State Models" in IEEE Trans. on Software Engineering, vol.17, No.6, June 1991.

[GONE70] GÖNENÇ, G. "A Method for Design of Fault-Detection Experiments" in IEEE Trans. on Computers, vol.C-19,No.6, 1970, pp 551-558.

[HENN64] HENNIE, F.C. "Fault Detecting Experiments for Sequential Circuits" in Proc.5th Ann. Symp. on Switching Theory and Logical Design Princeton University, Princeton, N.J., November 1964, pp 95-110.

[NAKA94a] NAKAZATO, K. K., MALDONADO, J. C., FABBRI, S. C.P.F. e MASIERO, P. C., "Propriedades de Máquinas de Estado Finito relevantes para Critérios de Geração de Seqüências de Teste", Relatório Técnico, Instituto de Ciências Matemáticas de São Carlos - USP, 1994 (em elaboração).

[NAKA94b] NAKAZATO, K. K., MALDONADO, J. C., FABBRI, S. C.P.F. e MASIERO, P. C. , "Seqüências Básicas de Máquinas de Estado Finito : Aspectos Teóricos e de Implementação.", Relatório Técnico, Instituto de Ciências Matemáticas de São Carlos - USP, 1994 (em elaboração).

[NAKA94c] NAKAZATO, K.K. "Módulo de Geração de Seqüências de Teste" Dissertação de Mestrado, ICMSC/USP,1994 (em elaboração).

[RAYN87] RAYNER, D. "Standardizing Conformance Testing for OSI" in Comput. Networks and ISDN Systems,vol.14 n.1,pp79-98,1987.

[SARI89] SARIKAYA, B. "Conformance Testing : Architecture and Test Sequences" in Comput. Networks and ISDN Systems, vol.17, pp111-126, 1989.

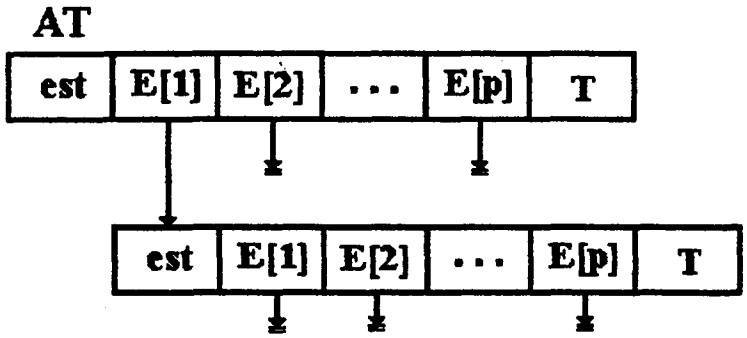
[SIDH89] SIDHU, D.P. and LEUNG, T.K. "Formal Methods for Protocol Testing : A Detailed Study." in IEEE Trans. on Software Engineering, vol.15, April 1989, pp 413-426.

Apêndice A

São descritas neste apêndice as principais estruturas de dados utilizadas nos algoritmos e procedimentos de implementação do critério W, além da descrição detalhada desses algoritmos.

A1. Árvore de Teste

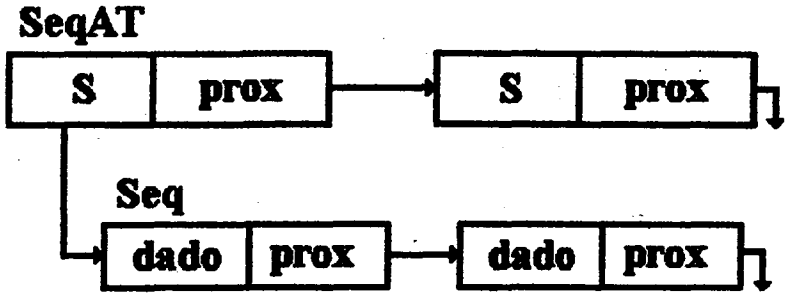
A Árvore de Teste da MEF M é representada pela estrutura dinâmica AT da seguinte forma :



onde :
est = identificação do estado que representa o nó de AT.
E[i] = ponteiro para o nó associado ao símbolo de entrada X[i].
T = rótulo de identificação de um nó terminal.

A2. Lista de Caminhos da Árvore de Teste

Esta lista que representa todos os caminhos da Árvore de Teste é uma estrutura dinâmica denominada SeqAT :



Define-se a seguir a notação da estrutura SeqAT e a estrutura Seq incluída em SeqAT :

1) Estrutura SeqAT :

S = ponteiro para uma lista que representa um caminho da Árvore de Teste.

prox = ponteiro para o próximo caminho da Árvore de Teste.

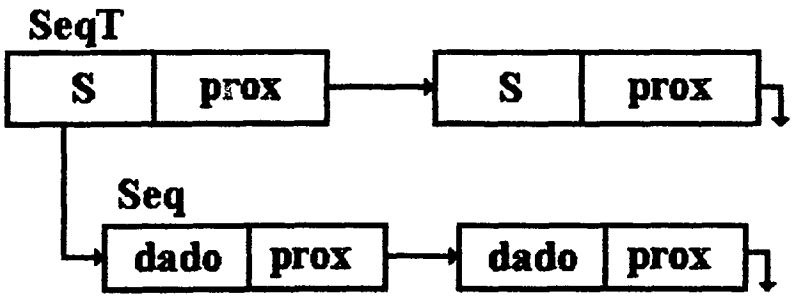
2) Estrutura Seq :

dado = estado da MEF M.

prox = ponteiro para o próximo estado da MEF M que faz parte do caminho da Árvore de Teste.

A3. Lista das Seqüências de Teste geradas pelo critério W

Esta lista representa todas as seqüências de teste geradas de acordo com o critério W e se apresenta como uma estrutura dinâmica denominada SeqT da seguinte forma :



Define-se a seguir a notação das estruturas SeqT e Seq :

1) Estrutura SeqT

S = ponteiro para uma lista que representa uma seqüência gerada pelo critério W.

prox = ponteiro para a próxima seqüência gerada pelo critério W.

2) Seq :

dado = símbolo de entrada que faz parte da seqüência gerada pelo critério W.

prox = ponteiro para o próximo símbolo de entrada da seqüência gerada pelo critério W.

A4. Geração da Árvore de Teste

Este algoritmo gera a Árvore de Teste através do uso de recursão. Destacam-se os seguintes aspectos:

1) Parâmetro de Entrada :
Sub-Tabela Sv+1 da Máquina M representado por FS1.

2) Parâmetro de Saída :
Árvore de Teste AT.

3) Descrição das Variáveis Locais :
AT1: nó gerado por AT.
RAIZ : raiz da Árvore de Teste.
NT : variável auxiliar para os nós gerados por AT.
I,J : contador de laço.
AUX2 : variável auxiliar.
VAZIA : variável dinâmica não inicializada.

4) Procedimentos Utilizados :
BuscaT.

5) Procedimentos que o utilizam :
Nenhum.

6) Descrição do Algoritmo em Pseudocódigo :

```
Procedimento ConstroiAT(FS1,AT)
início
  se AT=VAZIA
     então
       /* Geração do nó raiz da Árvore de Teste  */
       Aloque espaço para AT
       RAIZ:=AT
       para I:=1 até P faça
          AT.E[I]:=VAZIA
       fim para
       AT.T:=0
       AT.est:=S[1]
     fim se
  para I:=1 até P faça
     Aloque espaço para AT1
     AT.E[I]:=AT1
     para J:=1 até P faça
        AT1.E[J]:=VAZIA
     fim para
     AT1.T:=0
     AUX2:=AT.est
     AT1.est:=FS1[AUX2,I]
  fim para
  I:=0
  enquanto I<P
     NT:=AT.E[I]
     NT.T:=0
     se BuscaT(RAIZ,NT)
        então  NT.T:=1
     fim se
     I=I+1
  fim enquanto
  /*  Geração do próximo nível da Árvore de Distinção  */
  para I:=1 até P faça
     NT:=AT.E[I]
     se NT.T = 0
       então ConstroiAT(FZ1,FS1,NT)
     fim se
  fim para
fim ConstroiAT
```

A5. Busca de um Nó na Árvore de Teste

Este procedimento faz uma busca na Árvore de Teste com o objetivo de identificar se um nó já foi ou não incluído nesta árvore.

1) Parâmetro de Entrada :

Raiz da Árvore de Teste representada por RAIZT.

2) Parâmetro de Saída :
Nenhum.

3) Descrição das Variáveis Locais :
AT1: nó gerado por AT.
RAIZ : raiz da Árvore de Teste.
NT : variável auxiliar para os nós gerados por AT.
I,J : contador de laço.
PARADA,
CONTINUA,
AUX,
AUX1,
ANO,
AJUDA1,
PRIMEIRO,
INDICEAUX : variáveis auxiliares.
VAZIA : variável dinâmica não inicializada.

4) Procedimentos Utilizados :
Nenhum.

5) Procedimentos que o utilizam :
ConstroiAT.

6) Descrição do Algoritmo em Pseudocódigo :

```
Procedimento BuscaT(RAIZT,NT)
início
  I:=1    J:=1
  CONTINUA:=      PARADA:=0
  AUX:=RAIZT
  se RAIZT.est=NT.est
    então CONTINUA:=0
  enquanto I<P e CONTINUA=1
    se RAIZT.E[I] ≠ VAZIA
      se RAIZT.E[I]=NT
        J:=0      PARADA:=1
        CONTINUA:=0
        PRIMEIRO := 1
        enquanto J < P e PARADA=1
          se RAIZT.E[J].est=NT.est e PRIMEIRO=1
            então
              INDICEAUX:=J
              PRIMEIRO:=0
          fim se
          J:=J+1
        fim enquanto
        se INDICEAUX=I
          então PARADA:=0
        fim se
      fim se
      se RAIZT.E[I].est=NT.est e RAIZT.E[I] ≠ NT
        então PARADA:=1
      fim se
    fim se
    I:=I+1
  fim enquanto
  I:=0
  se RAIZT ≠ NT e   CONTINUA = 1
    então
      AUX:=RAIZT
      AUX1:=AUX.est
      AJUDA1:=NT.est
      se AUX1=AJUDA1
        então PARADA:=1
      se não PARADA
        então
          ANO:=AUX.E[I]
          enquanto não PARADA e I<P e ANO ≠ VAZIA
            se BuscaT(ANO,NT)
              então PARADA:=1
            I:=I+1
            ANO:=RAIZT.E[I]
          fim enquanto
      fim se
  fim se
  retorne(PARADA)
fim BuscaT
```

A6. Geração das Seqüências de Teste

Este algoritmo gera as seqüências de teste de acordo com o critério W. Desta forma destacam-se os seguintes aspectos :

1) Parâmetros de Entrada :
Conjunto de Caracterização representado por SeqW.
Lista de caminhos da Árvore de Teste da Máquina M representado por SeqAT.

2) Parâmetro de Saída :
Lista de seqüências de teste representado por SeqT.

3) Descrição das Variáveis Locais :
AUX1 : variável auxiliar para manipular a Lista de caminhos da Árvore de Teste.
AUX2 : variável auxiliar para manipular a Lista de seqüências do Conjunto de Caracterização W.
AUX3,
AUX3A : variáveis auxiliares para manipular a Lista de seqüências de teste geradas pelo critério W.
VAZIA : variável dinâmica não inicializada.

4) Procedimentos Utilizados :
Concat.

5) Procedimentos que o utilizam :
Nenhum.

6) Descrição do Algoritmo em Pseudocódigo :

```
GeraçãoST(SeqW,SeqAT,SeqT)
início
AUX1:=SeqAT
AUX3:=SeqT
enquanto AUX1 ≠ VAZIA
   AUX2:=SeqW
   enquanto AUX2 ≠ VAZIA
      se AUX3=VAZIA
         então
            Aloque espaço para AUX3
            Concat(AUX1.S,AUX2.S,AUX3.S)
            AUX3.prox:=VAZIA
         senão
            AUX3A:=AUX3
            enquanto AUX3 ≠ VAZIA
               AUX3A:=AUX3
               AUX3:=AUX3.prox
            fim enquanto
            Aloque espaço para AUX3
            AUX3A.prox:=AUX3
            Concat(AUX1.S,AUX2.S,AUX3.S)
      fim se
      AUX2:=AUX2.S
   fim enquanto
   AUX1:=AUX1.S
fim enquanto
fim GeraçãoST
```

A7. Concatenação da Árvore de Teste com o Conjunto de Caracterização W.

Este algoritmo concatena um caminho da Árvore de Teste com uma seqüência que faz parte do Conjunto de Caracterização W. Assim os seguintes aspectos devem ser destacados :

1) Parâmetros de Entrada :

Seqüência do Conjunto de Caracterização representado por Seq1.
Caminho da Lista de caminhos da Árvore de Teste da Máquina M representado por Seq2.

2) Parâmetro de Saída :

Seqüência de teste representada por Seq3.

3) Descrição das Variáveis Locais :

AUX1,
AUX1A : variáveis auxiliares para manipular a lista de estados do caminho da Árvore de Teste.
VAZIA : variável dinâmica não inicializada.

4) Procedimentos Utilizados :

Nenhum.

5) Procedimentos que o utilizam :

GeraçãoST.

6) Descrição do Algoritmo em Pseudocódigo :

```
Concat(Seq1Seq2,Seq3)
início
AUX1:=Seq1
AUX1A:=AUX1
enquanto AUX1 ≠ VAZIA
   AUX1A:=AUX1
   AUX1:=AUX1.prox
fim enquanto
AUX1A.prox:=SEQ2
SEQ3:=AUX1
fim Concat
```