

Testes baseados na especificação

Modelos de estado

Tópicos

- *Modelo de estados: apresentação*
- *Características*
- *Propriedades*
- *Testes de transição de estados*

Tópicos

- *Modelo de estados: apresentação*
- *Características*
- *Propriedades*
- *Testes de transição de estados*

Máquinas de estado

- *Máquinas Finitas de Estado (MFE)*
 - *aplicação na Engenharia (hw e sw) de modelo matemático denominado **autômato finito***
 - *são usadas desde os anos 50 para modelar circuitos*
 - *usadas desde os anos 60 para modelar sw*
 - *protocolos de comunicação*
 - *analísadores de sintaxe*
 - *sistemas de controle*
 - *interfaces-usuário*
 - *comportamento de objetos*

MFE considerada

- *Modelo de Mealy*
 - *Máquina hipotética que pode estar em um único estado, dentre um conjunto finito de estados possíveis, em um determinado momento.*
 - *A máquina pode mudar de estado em resposta a uma entrada, podendo produzir uma saída.*
 - *Tanto o novo estado quanto a saída são funções unicamente do estado atual e da entrada fornecida.*

Elementos de uma MFE

- *Estado*
 - *conjunto de valores dos dados do sistema em um determinado momento*
- *Transição*
 - *leva o sistema de um estado para outro devido à ocorrência de um evento*
- *Evento*
 - *entrada ou período de tempo*
- *Ação*
 - *resultado ou saída produzida em resposta ao evento ocorrido*

Elementos de uma MFE (cont.)

- *Estado inicial*
 - *estado do sistema (ou componente) em que o 1º evento é aceito*
- *Estado origem / estado destino*
 - *uma transição leva o sistema de um estado origem a um estado destino, os quais podem ser iguais*
- *Estado atual*
 - *estado corrente em que se encontra a execução do sistema*
- *Estado final*
 - *estado do sistema no qual eventos não são mais aceitos. O sistema pode ter 0 ou mais estados finais.*

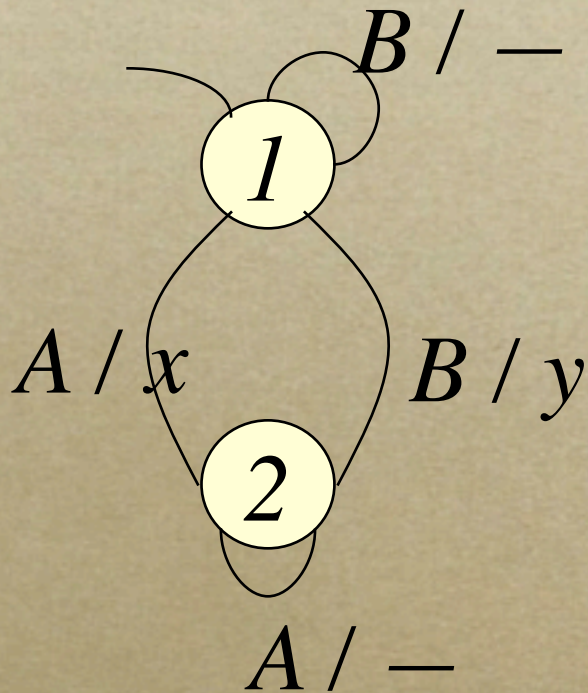
Semântica

- *A máquina inicia no estado inicial*
- *A máquina espera por um evento durante um tempo indeterminado*
- *A máquina recebe um evento*
- *Se o evento não é aceito no estado corrente da máquina, ele é ignorado*
- *Se o evento é aceito no estado atual: a transição correspondente é disparada, a ação associada é ativada e o estado designado como próximo torna-se o estado atual (pode ser o mesmo)*
- *Os passos anteriores se repetem até que a máquina chegue a um estado final*

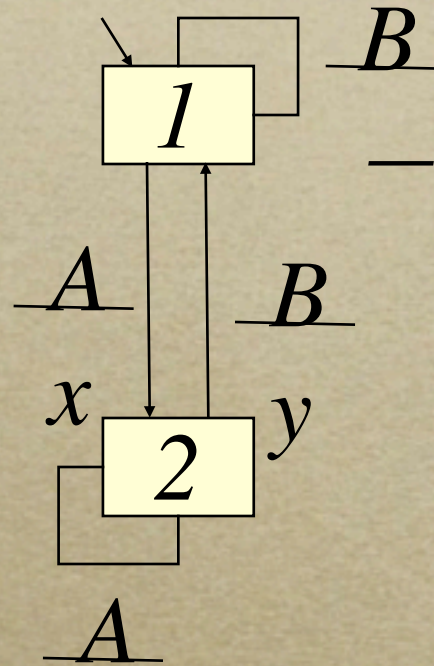
Um modelo de estados ...

- *não leva em conta a maneira pela qual um evento é produzido*
- *trata um evento por vez. Uma única transição pode ser disparada em um dado momento*
- *não aceita nenhum evento além daqueles especificados*
- *só pode estar em um único estado em um dado momento*
- *é estático: estados, eventos, transições e ações não podem ser criados nem removidos quando a máquina é executada*
- *não descreve como uma ação produz uma saída*
- *não tem intervalo de tempo associado a nenhum aspecto do modelo. O disparo de transições é considerado atômico, ie, não consome tempo*

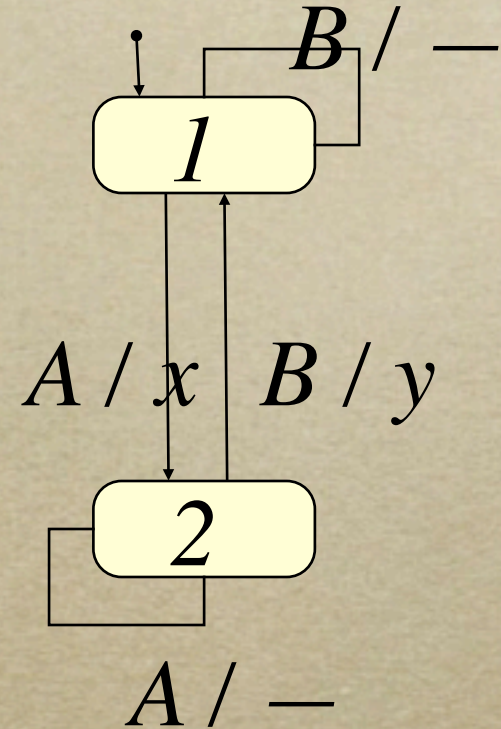
Notação 1: diagrama de estado



clássico



**Análise
Estruturada**



UML

Notação 2: tabela de transição

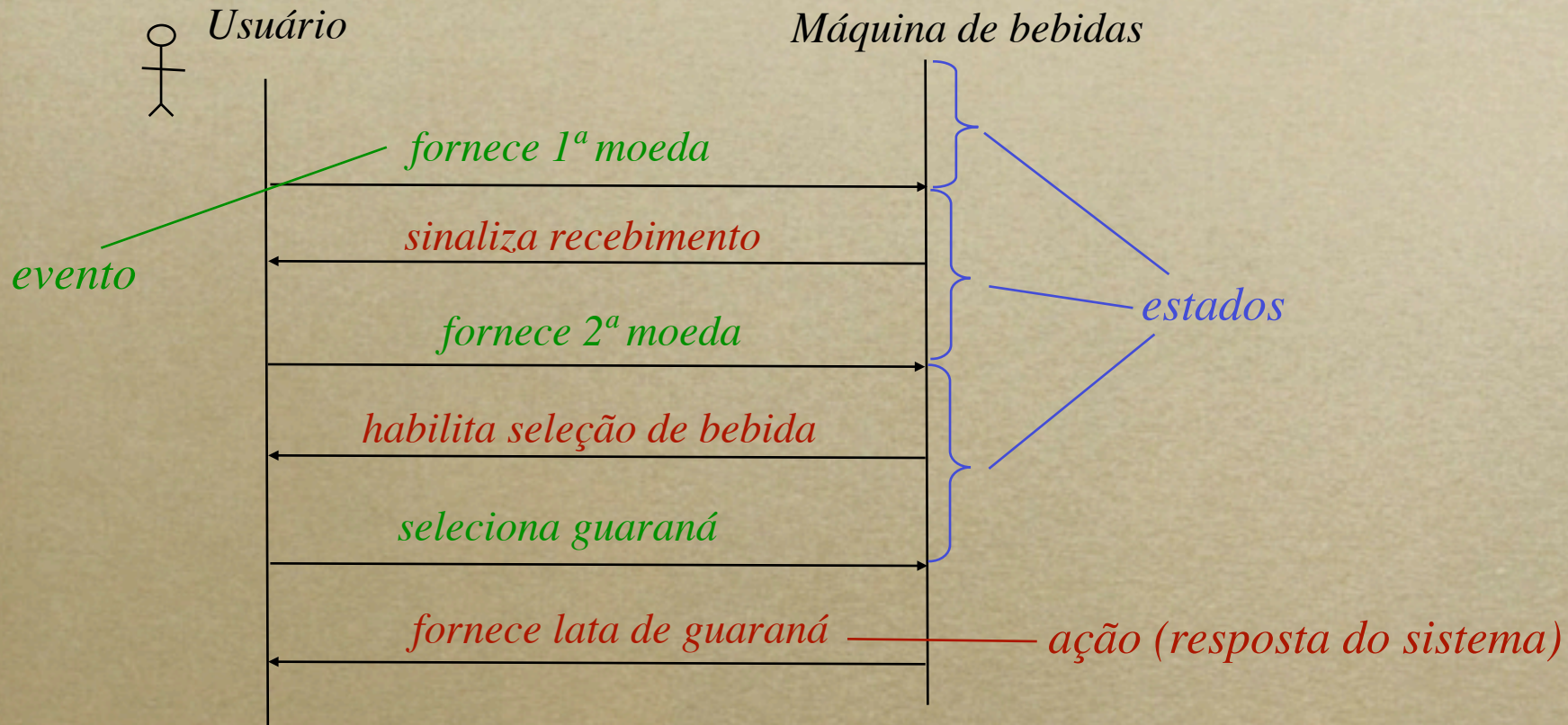
Estados	Eventos	
	<i>A</i>	<i>B</i>
<i>1</i>	<i>2/x</i>	<i>1/-</i>
<i>2</i>	<i>2/-</i>	<i>1/y</i>

estado origem *estado destino* *ação*

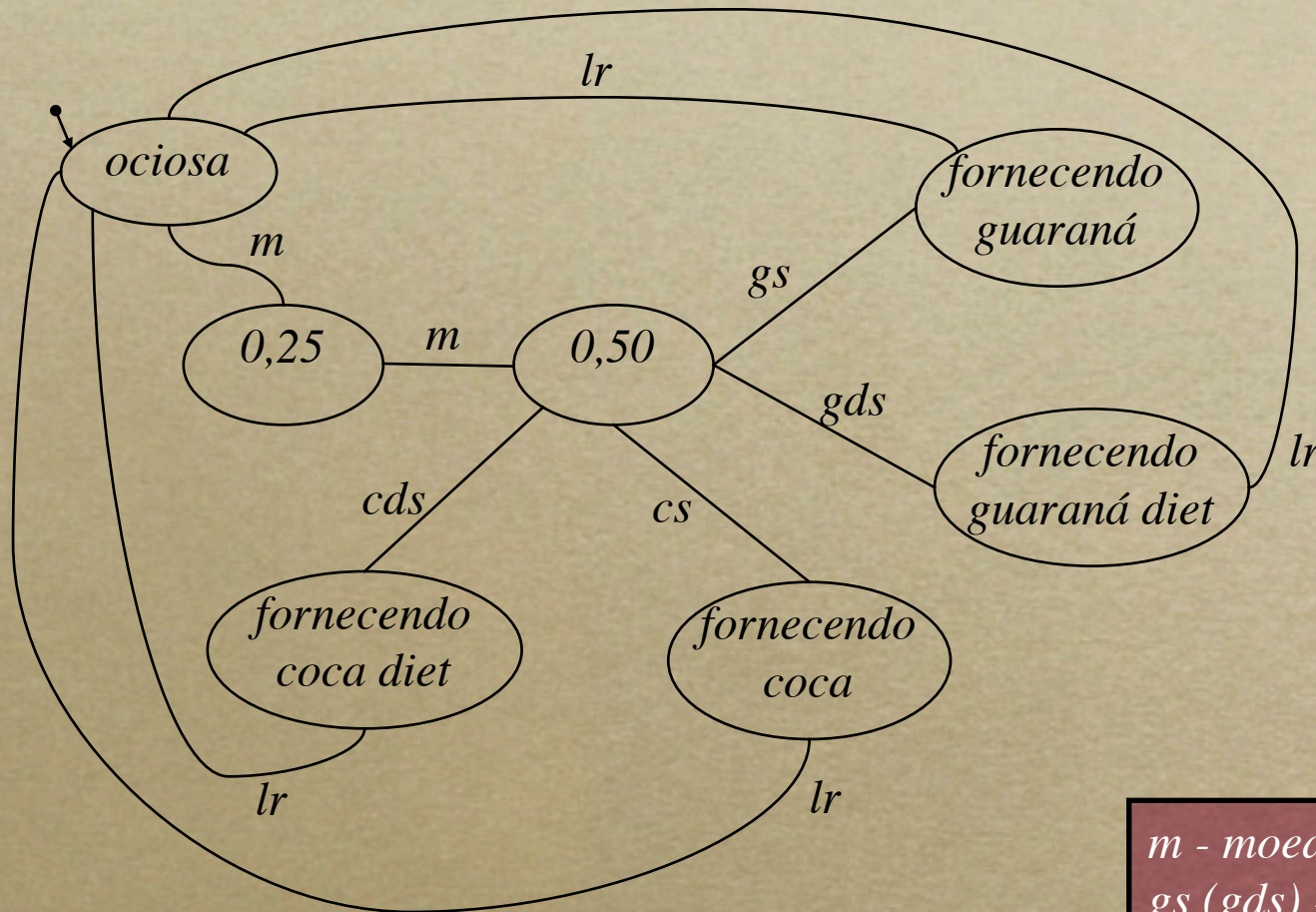
Exemplo: máquina de venda de refrigerantes (versão 1.0)

- *Suponha que a máquina funcione da seguinte forma:*
 - *assim que é ligada, a máquina aguarda moedas*
 - *um refrigerante custa 0,50. A máquina aceita somente moedas de 0,25*
 - *após receber as duas moedas, a máquina aguarda o usuário selecionar uma das seguintes bebidas: “guaraná”, “guaraná diet”, “coca” ou “coca diet”*
 - *quando o usuário seleciona a bebida a máquina fornece a lata*
 - *depois que o usuário pega a lata, a máquina volta a esperar moedas*

Exemplo: identificando eventos e estados



Modelo de estados (versão 1.0)



m - moeda 0,25 fornecida
gs (*gds*) - guaraná (*diet*) selecionado
cs (*cds*) - coca (*diet*) selecionado
lr - lata retirada

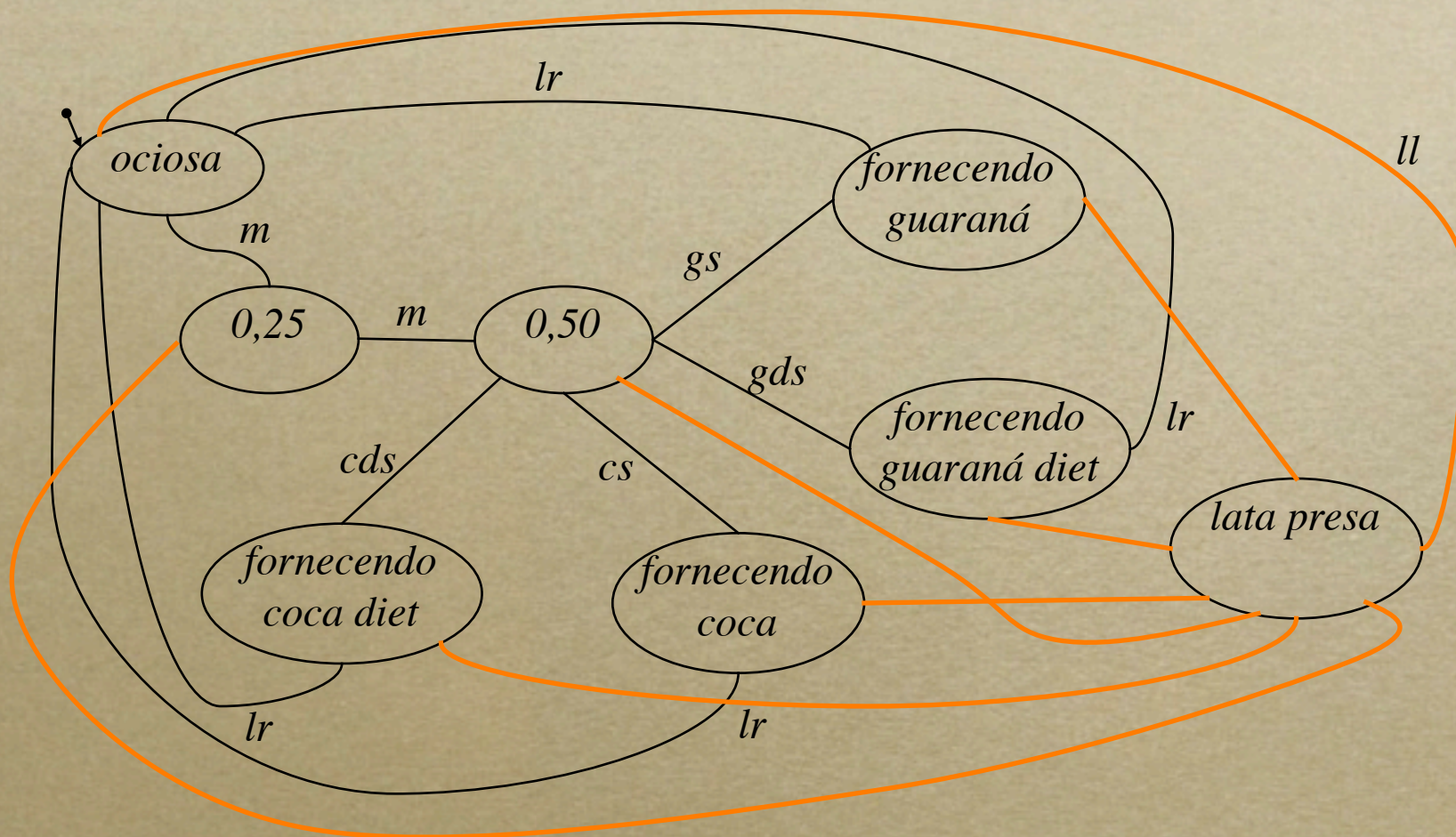
Limitações do modelo básico

- *Escalabilidade limitada:*
 - *as máquinas são finitas, ou seja, os valores possíveis para as entradas devem ser finitos*
 - *n° de estados tem tendência a aumentar muito na medida em que aumenta o nível de detalhe do que se deseja especificar (problema de explosão de estados)*

Exemplo - outra máquina de venda de refrigerantes (versão 2.0)

- *Latas podem ficar presas na máquina:*
 - *o sistema é notificado automaticamente de que há lata presa*
 - *neste caso a máquina não deve mais aceitar moedas e nem fornecer outro refrigerante até que a lata seja liberada*
 - *após liberar a lata, a máquina volta a aceitar moedas*
- *Mudanças no modelo:*
 - *quantos estados a mais ?*
 - *Quantas transições a mais ?*

Modelo de estados (versão 2.0)



ll - lata liberada

Exemplo - outra máquina de venda de refrigerantes (versão 2.1)

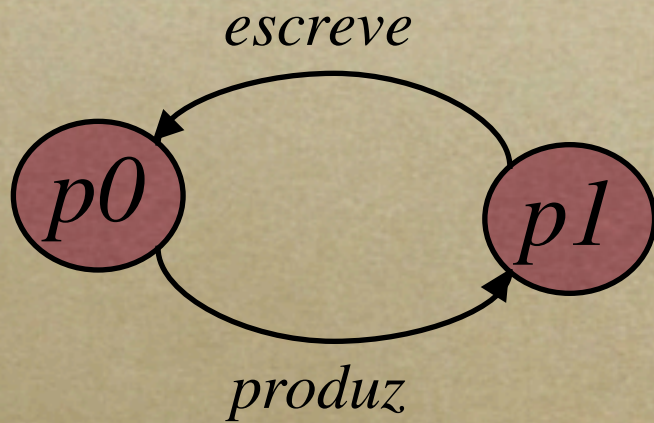
- *Latas podem ficar presas na máquina:*
 - *o sistema é notificado automaticamente de que há lata presa*
 - *neste caso a máquina não deve mais aceitar moedas e nem fornecer outro refrigerante até que a lata seja liberada*
 - *após liberar a lata, a máquina volta ao mesmo estado em que estava antes de a lata ficar presa*
- *Mudanças no modelo:*
 - *quantos estados a mais ?*
 - *Quantas transições a mais ?*

Limitações do modelo básico

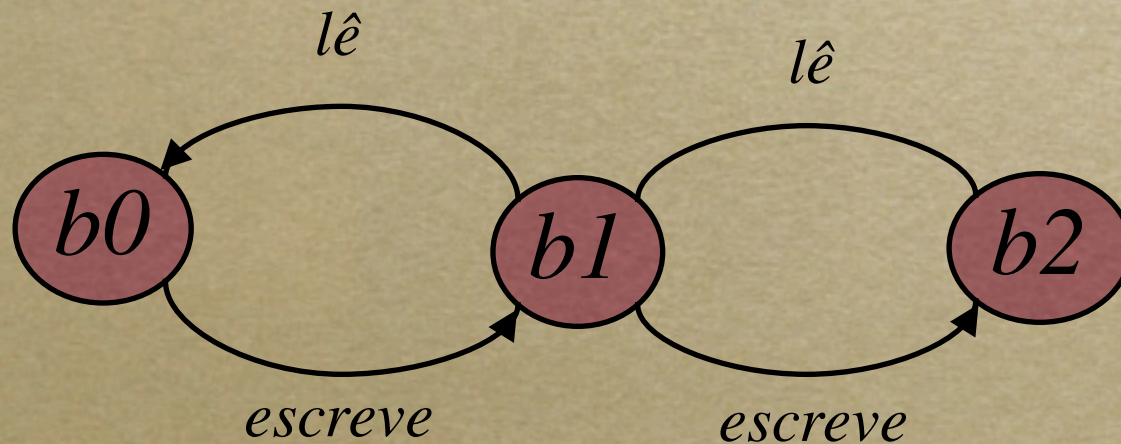
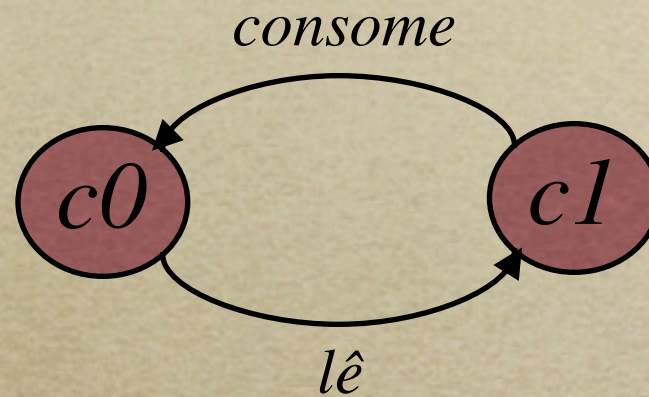
- *Concorrência:*
 - *dado que a máquina só pode estar em um estado por vez, o modelo não permite representar o disparo simultâneo de mais de uma transição, ou a chegada simultânea de mais de um evento (modelo síncrono)*
 - *concorrência é representada pela máquina produto, cujos estados são combinações dos estados de 2 ou mais máquinas básicas \Rightarrow risco de explosão de estados*

Exemplo: problema do produtor-consumidor

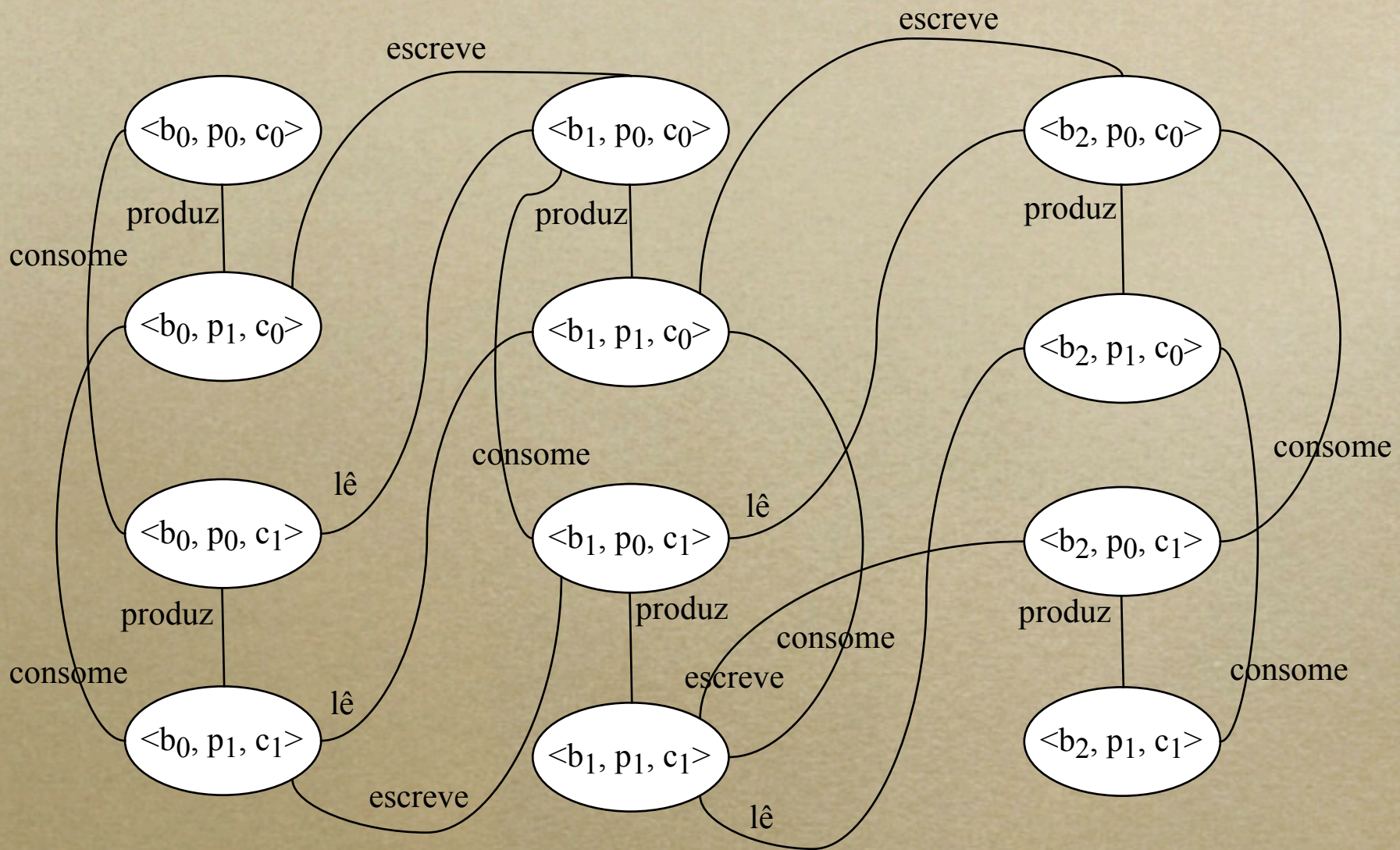
Produtor



Consumidor



Buffer



Propriedades

- *Para ser testável uma MFE deve ser*
- *Completa: a cada par evento-estado existe uma transição*
- *Mínima: não possui estados equivalentes*
- *Fracamente conexa: todo estado é alcançável a partir do estado inicial*
- *Determinista: para cada estado existe uma e somente uma ação*

Dois estados i e j são equivalentes se todas as seqüências de eventos aplicadas aos dois estados resultam nas mesmas ações

Um estado j é alcançável a partir de outro estado i quando existe pelo menos uma seqüência válida de eventos que leve a MFE de i a j .

Propriedades

- *Uma MFE interessante para testes deve ser:*
 - *Completa: a cada par evento-estado deve ser associada uma transição*
 - *Mínima: não possui estados equivalentes*
 - *Inicialmente conexa: todo estado é alcançável a partir do estado inicial*
 - *Determinista: a cada par evento-estado está associada uma e somente uma transição*

Passos para os testes baseados em MFE

- *Construir o modelo de estados*
- *Verificar o modelo construído*
- *Gerar os casos de testes obtendo as entradas e saídas esperadas*
- *Executar os testes gerados*
- *Analisar os resultados:*
 - *saída observada = saída esperada ?*
 - *estado final correto ?*

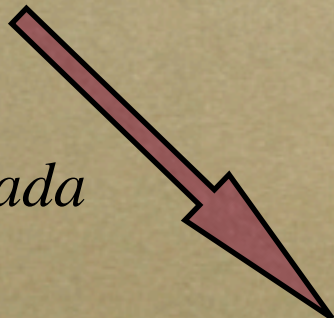
Geração de testes: princípio

Implementação

↓ *entrada*

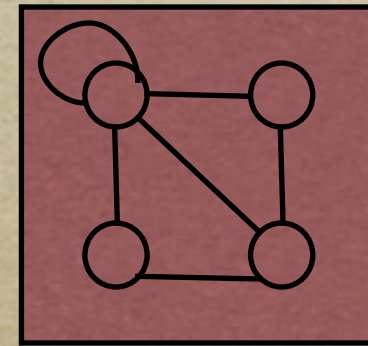


saída esperada

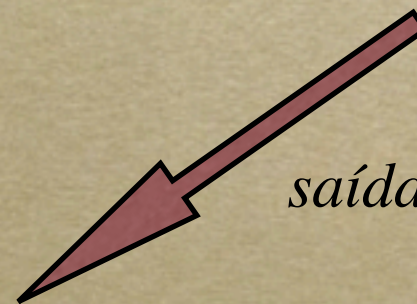


Especificação

entrada ↓



saída observada



se diferentes \Rightarrow erro

Geração de testes: método T

- *Critério: exercitar cada transição $t = \langle sm, sn \rangle$ pelo menos 1 vez*
 - *Caso de teste é formado por:*
 - *entradas que levem I do estado inicial s_0 ao estado sm ;*
 - *entrada x_m , que ativa a transição $\langle sm, sn \rangle$*
 - *entradas que retornem I ao estado inicial*
- *Não verifica se estado final correto*
- *Simples de implementar*

Exercício 1

- *Prepare casos de testes para o sistema de máquina de venda de bebidas.*

Exercício 2

- *o sistema solicita o nome do usuário (user name)*
- *se o nome não é reconhecido pelo sistema, este solicita ao usuário que o forneça novamente, até que um nome válido seja fornecido*
- *o sistema solicita a senha*
- *se a senha é incorreta, o usuário tem uma chance a mais para fornece-la. Se ambas as tentativas falharem, o usuário deve recomeçar todo o processo*
- *se a senha é válida, o sistema solicita o nível de segurança da sessão a ser iniciada*

Exercício 3

- *Forneça a especificação na forma de modelo de estados para um sistema de controle de aquecimento de uma casa, composto de um aquecedor, um termostato e um ventilador. O controle da temperatura é distribuído, ie, cada cômodo tem um controlador de temperatura. Se a temperatura em um cômodo cai abaixo de $t-2$, onde t é a temperatura desejada, o aquecedor é ligado. Quando a temperatura do aquecedor atinge um limite máximo T , este é desligado e o ventilador é acionado para espalhar o ar quente. Enquanto isso o termostato monitora e registra a temperatura do ambiente. Quando esta atinge $t+2$, o aquecedor é desligado. O ventilador continua ligado até que a temperatura do aquecedor chegue a $T-5$. Assuma que $t+2 > T-5$.*

