

# Teste em contexto: viável do ponto de vista prático ou meramente uma teoria?

Jurandy Martins Soares Júnior  
(Mestrando)

Prof. Ricardo Anido – IC – Unicamp  
(Orientador)



## Agenda

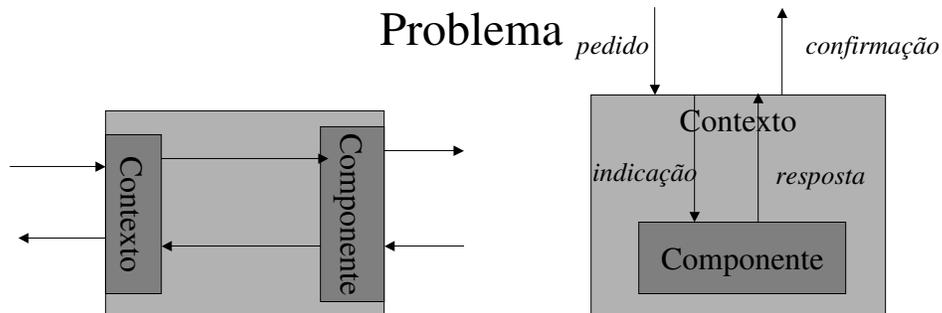
- Motivação
- Problema
- Formalização do problema
- Revisão de teste de conformidade
- Revisão de teste em contexto
- Mapeamento de seqüências de teste do componente em seqüências de teste do contexto
- Limitações de teste em contexto
- Conclusão

## Motivação

- Uso crescente de sistemas comunicantes em nossa tecnosfera
- Falta de métodos eficientes para testar componentes e/ou sistemas embutidos em outros
- Velocidade de desenvolvimento de dispositivos
- Teste de interoperabilidade entre camadas



## Problema

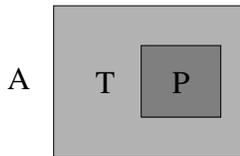


- Questões:
  - Quão testável é o componente a partir do contexto?
  - Qual a flexibilidade do contexto no que se refere à testabilidade de componentes?
  - Quanto de cobertura de falhas oferece o contexto?

## Formalização do Problema

Definição 1 - A noção de máquina de estados finitos com entradas e saídas é uma quintupla  $\langle S, I, O, \delta, \lambda, s_0 \rangle$ , tal que:

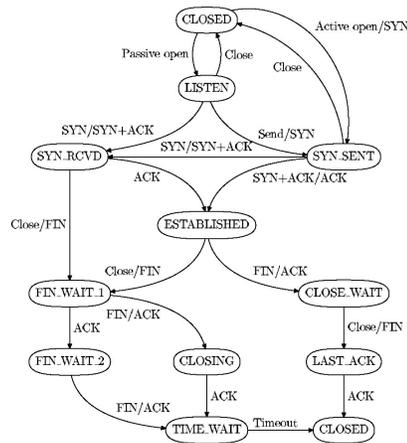
- (1)  $S, I$  e  $O$  são conjuntos finitos, onde  
 $S$  é um conjunto de estados;  
 $I$  é um conjunto de sinais de entrada, também conhecido como alfabeto de entrada;  
 $O$  é um conjunto de sinais de saída, também conhecido como alfabeto de saída.
- (2)  $\delta : S \times I \rightarrow 2^S$  é a função de transição de estado.
- (3)  $\lambda : S \times I \rightarrow 2^O$  é a função de saída.
- (4)  $s_0$  é o estado inicial.



## Modelos de falhas

- **Falha de saída:**
  - a IST possui uma se, para o estado correspondente e a entrada recebida, a IST produzir uma saída diferente da existente na especificação;
- **Falha de transição:**
  - uma transição possui uma falha se, para o estado correspondente e a entrada recebida, a IST entrar em um estado diferente, que não aquele descrito na especificação;
- **Falha de transição ausente:**
  - uma IST tem uma transição ausente se, para o estado atual, não existir uma transição que aceite próximo sinal de entrada da seqüência de teste.

## Revisão de Teste de Conformidade



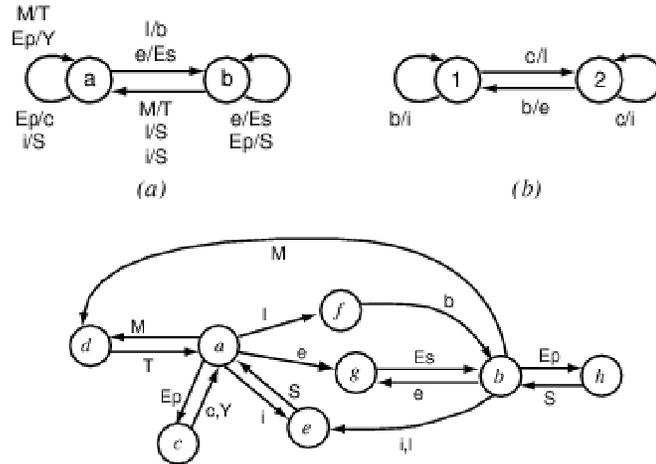
- (T) – passeio de transições
- (D) – seqüência de distinção
- (W) – conjuntos de diferenciação
- (U) – seqüências únicas de sinais de entrada e saída

Figura 2.4: Especificação do protocolo TCP

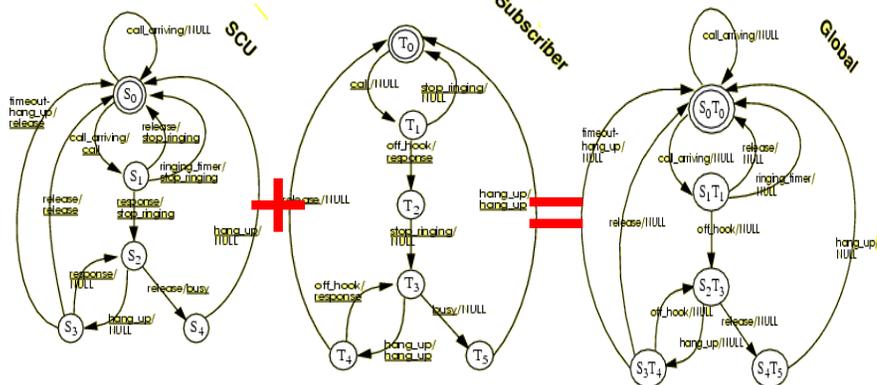
## Revisão de Teste de Conformidade em Contexto

On-the-fly	Jard, 1991
Produto cartesiano	Petrenko, 1992
Busca cega	Lee, 1996
Minimização	Cavalli, 1998
Minimização p/ MEFs	Anido, 1999
Busca cega guiada	Lee, 1999
MEFs temporizadas	Uya, 2002
Cobertura de falhas	Petrenko, 2002
X	
Cobertura da especificação	

## Composição de MEFs

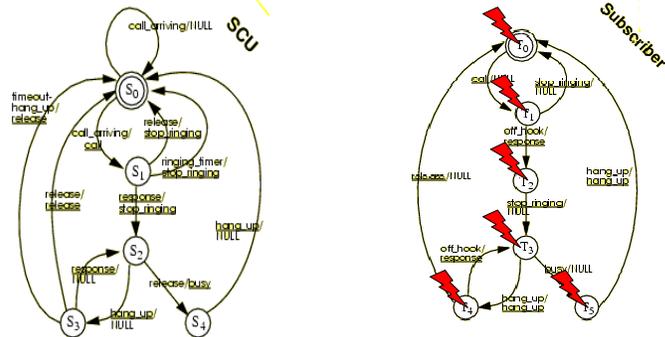


## Minimização



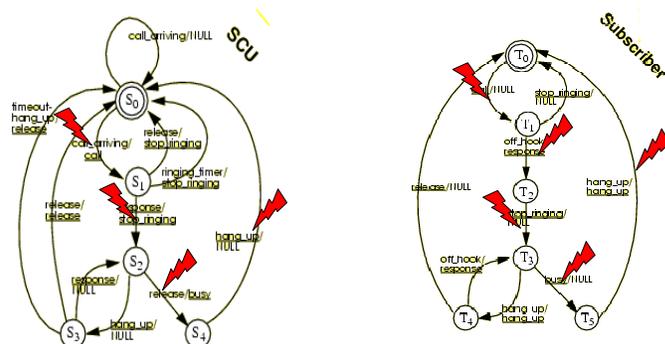
A MEF resultante apresenta alguns estados que nunca serão alcançados, e portanto, poderão ser removidos (Lima, 1997)

## Abordagem on-the-fly



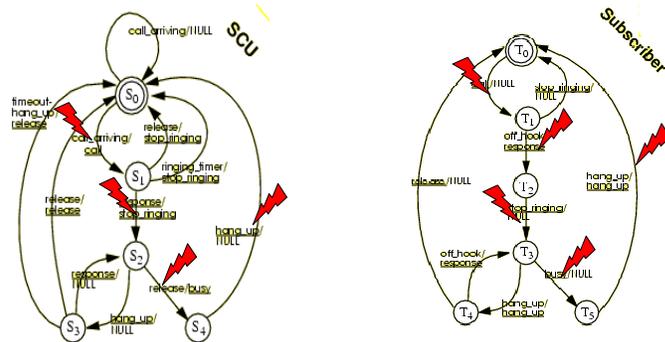
- “Um método intermediário ... baseado na busca em profundidade e ainda usa todo o espaço disponível para armazenar não só o caminho atual, mas também o maior número possível de estados já visitados” (Jard, 1991)

## Busca cega



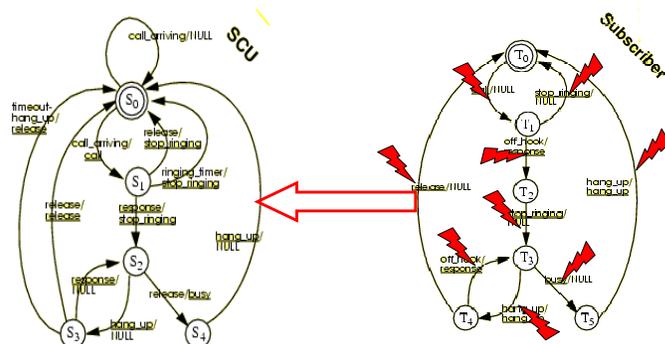
- “... se uma parte não testada do componente é encontrada (um alcance ou “hit”), mantém-se esta parte para a seqüência final de teste, e continua-se a busca a partir deste ponto.” (Lee, 1996)

## Busca cega “guiada” (“Hit-or-jump”)



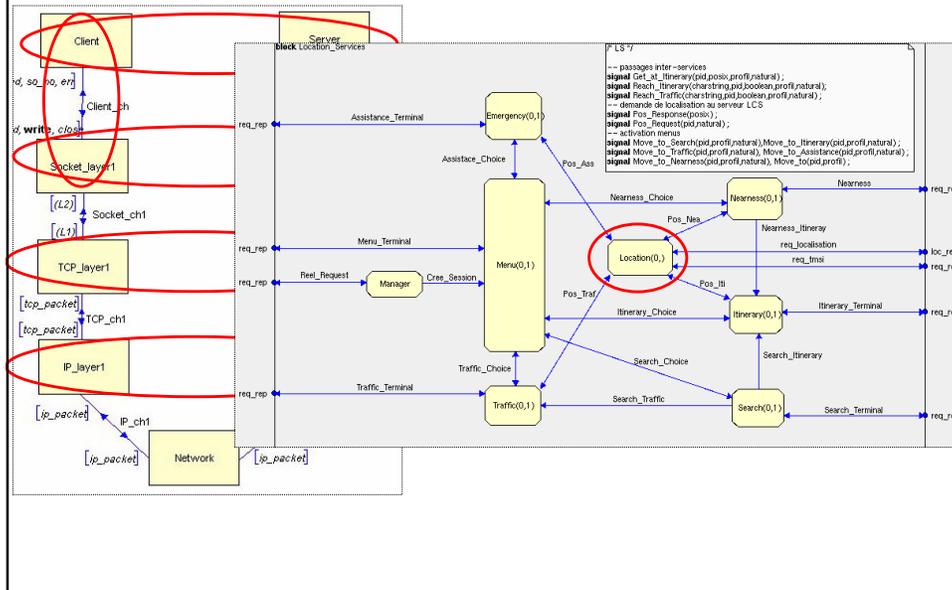
- “ ... o espaço necessário é determinado pelo usuário, como o limite de profundidade e a quantidade máxima de estados.”(Cavalli et Lee, 1998)

## Mapeamento de seqüências de teste do componente para o contexto



- Porque gerar cegamente seqüências de teste no contexto para exercitar o componente?
- Por que não fazer uma coisa de dentro para fora?

## Será viável???



## Conclusão

- Teste em contexto é parcialmente viável, tendo em vista que nem tudo no componente pode ser testado a partir do contexto, pois componente e contexto foram feitos para tratar de “tarefas distintas”, com unidades de dados distintas.
- Ademais, a testabilidade do componente a partir do contexto é limitada porque se eles tiverem sido bem desenvolvidos, eles por si serão altamente coesos e, entre si, fracamente acoplados.