

Aula Prática de ESD

(www.cin.ufpe.br/~acm/esd)

Usando FDR e Probe

Sidney Nogueira (scn@cin.ufpe.br)

Roteiro

- FDR e Probe
- Rodando ferramentas
- Exemplos e dicas de uso
 - Até Aula 3
 - Aulas 4 e 5

FDR e Probe

- FDR (Failures Divergence Refinement)
 - Verificador de modelos de CSP
 - Verifica em um processo a presença de
 - deadlock
 - livelock
 - não determinismo
 - Verifica refinamentos entre processos (traces, falhas e falhas e divergencias)
- Probe
 - Navegador de processos CSP
 - Permite o usuário simular o sistema e ver a progressão da especificação

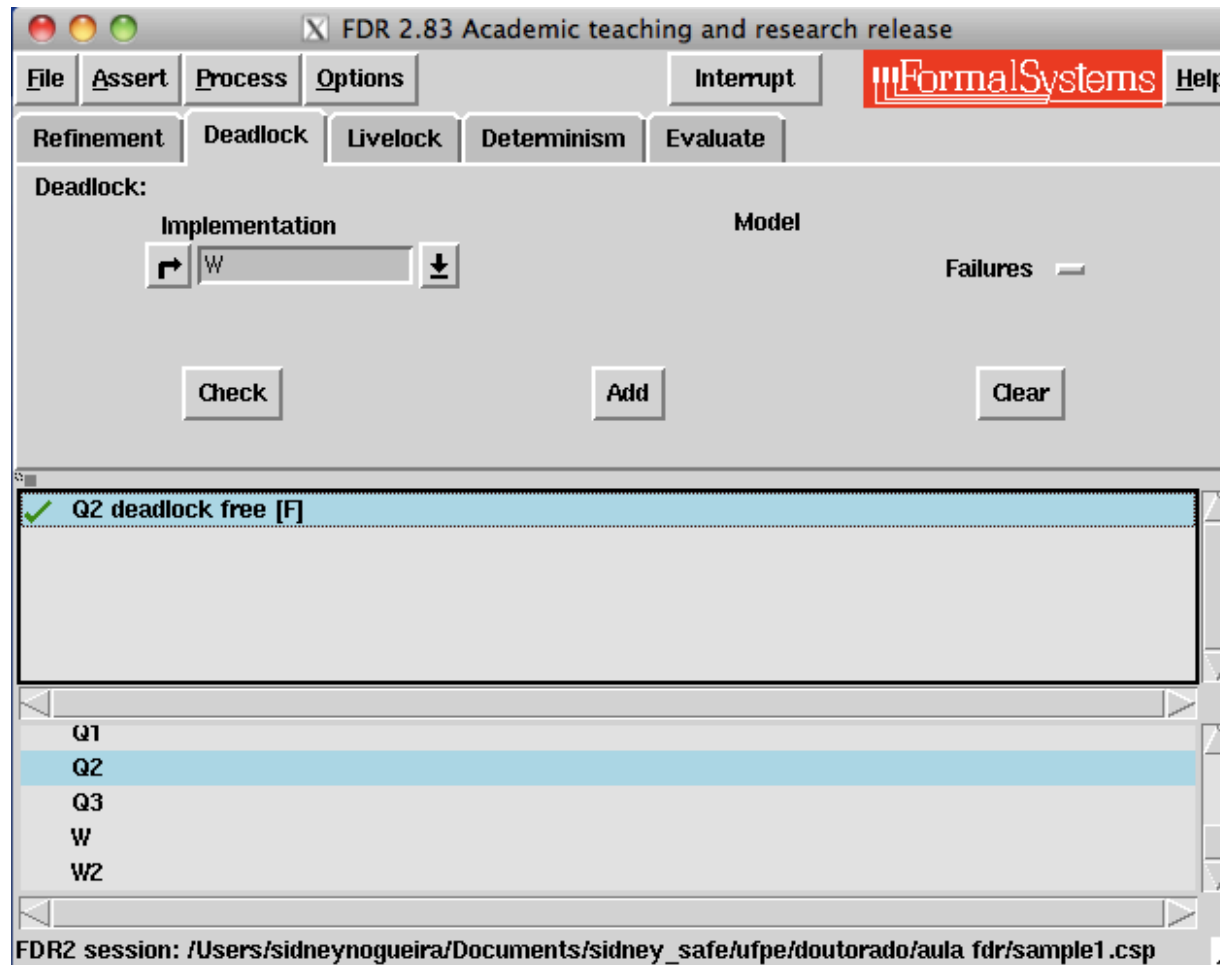
Antes de rodar

- Baixar de www.cin.ufpe.br/~scn/esd/
 - roteiro_aula_pratica.pptx (este arquivo)
 - sample1.csp
 - putty.exe
 - probe (executável para linux)
- Ter Xming instalado
- Ter permissão para logar no servidor altinho.cin.ufpe.br (linux)

Rodando FDR

- FDR (Failures Divergence Refinement)
 - Executar Xming
 - Executar e configurar putty
 - adicionar altinho.cin.ufpe.br no campo servidor
 - marcar SSH -> forward the X11
 - salvar configuração para não ter que repetir os 2 passos anteriores
 - Logar em altinho.cin.ufpe.br
 - Usuario/senha do windows
 - Depois de logar chamar comando “fdr2 &”
 - Verificar se o diretório remote está mapeado para sua conta (drive h:)
 - Para usar de casa é preciso estar na VPN

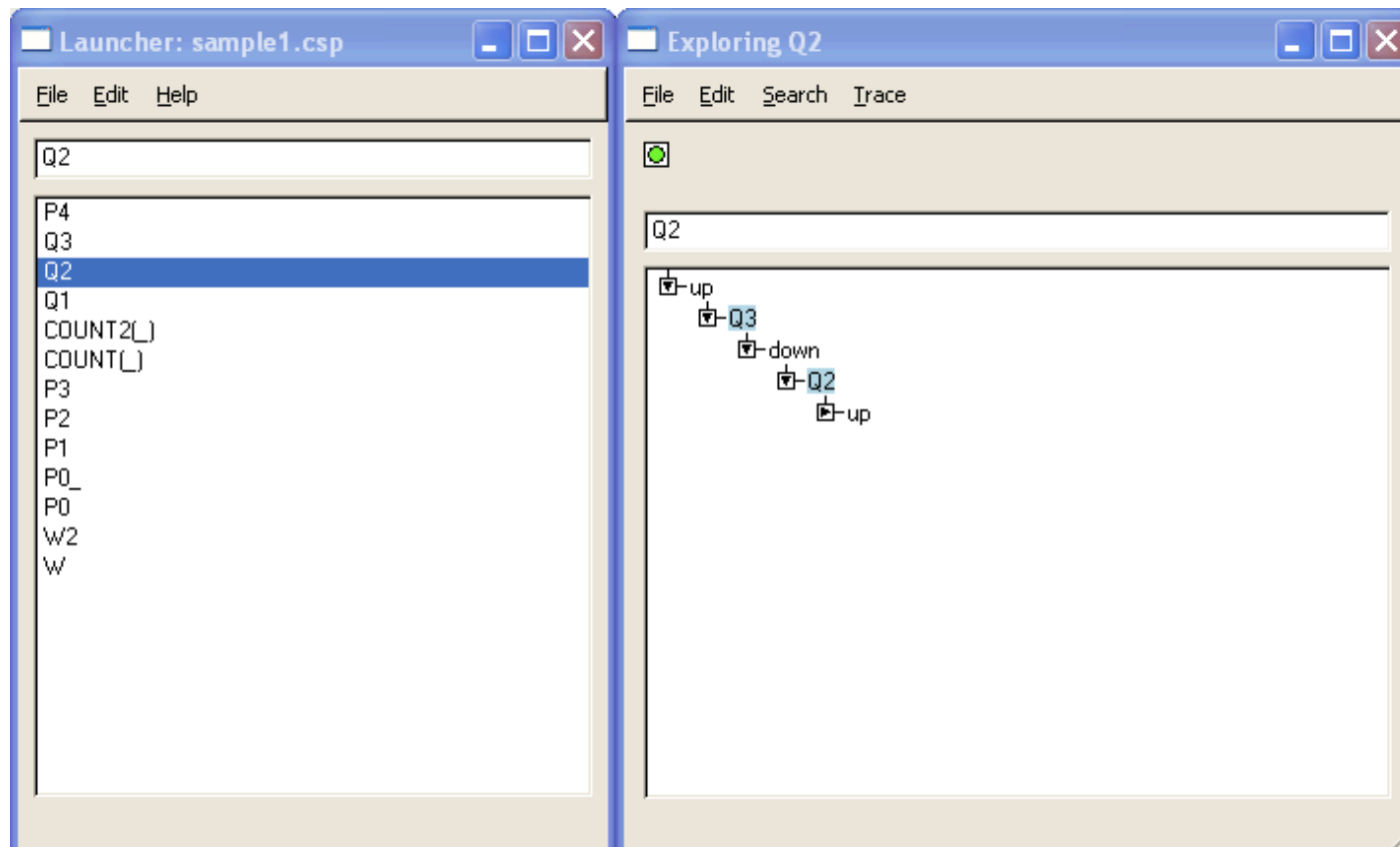
Tela do FDR



Rodando Probe

- Probe
 - copiar o arquivo executável (probe) da sua conta para o local em altinho. Considere dir o diretório em h: onde foi baixado o arquivo
 - `cp ~/remote/dir/probe .`
 - chamar pela linha de comando `./probe &`
 - diferente de FDR, tem versão para windows

Tela de Probe



Basico da interface de FDR

- File (Load, Reload)
- Lista de verificações (assertions)
- Lista de processos do arquivo carregado
- Janela de Mensagens
 - Mostrar: Options -> Show status
 - Detalhes da janela: Messages -> ...

Deadlock, deterministic, evaluate

- Processo W2
 - Especificação de canal restrita
 - Verificar deadlock
 - Contra exemplos
 - Expandir com Probe
 - Evaluate $\{|ch|\}$
- Processo W
 - Canal do tipo inteiro
 - Falha ao rodar no FDR (cardinalidade infinita)

Deadlock de Stop vs deadlock de Skip

- Processos P0 e P0_
 - Incluindo assert no .csp
 - Ex: assert P0 :[deadlock free [F]]
 - Diferenciando SKIP de STOP na verificação de deadlock de P0 e P0_
 - Evento tick

Processo parametrizado

- COUNT(n)
 - Problema ao verificar no FDR
 - Expansão ilimitada
 - Probe permite expandir por demanda
- COUNT2(n)
 - Ok para FDR e Probe
 - Limites superiores e inferiores

Busca no Probe

- Processo ATM1
 - Navegar no Probe
 - Buscar evento no probe (search -> Event)
 - Comando print, exs
 - `print { PIN.c | c <- CARD, c > 3, c < 7 }`
 - `print member(100, WA)`
 - `print head(<1,2>)`

Analizando determinismo

- Processos P1, P2 e P3
 - Analisar determinismo com FDR
 - Ex: `assert P1 :[deterministic [FD]]`
 - Em que ponto P2 é não determinístico? e P3?
Porque?
 - Use o probe para debugar

Typechecker

- FDR e Probe só encontram problemas nos tipos quando expandindo processos
- Typechecker detecta problemas de tipos varrendo estaticamente a especificação
 - a mensagem de erro é mais completa
- Disponível em www.fsel.com
 - Roda pela linha de comando
 - ex: 'checker sample1.csp'

Comparando processos

- Em FDR é possível analisar se um processo P equivale a Q se as verificações que seguem são verdadeiras
 - $\text{assert } P \text{ [FD= } Q$
 - $\text{assert } Q \text{ [FD= } P$
- Se alguma delas falha os processos não são equivalentes
- Isto pode ser feito em T, F ou FD

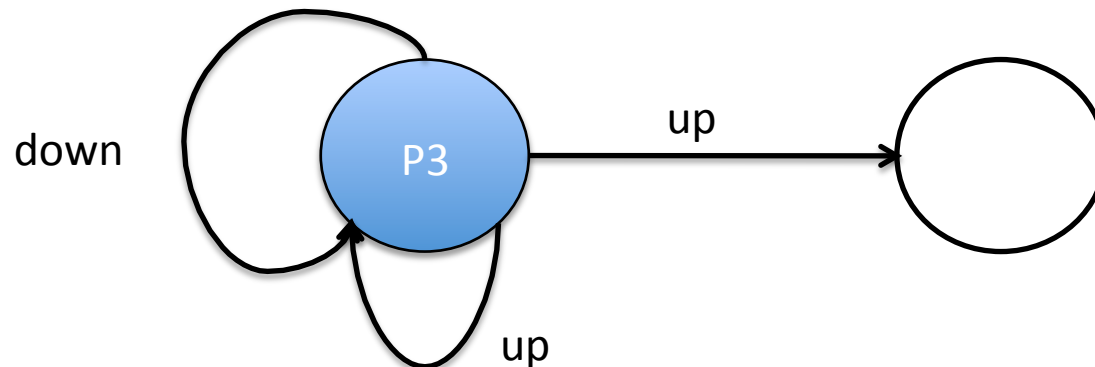
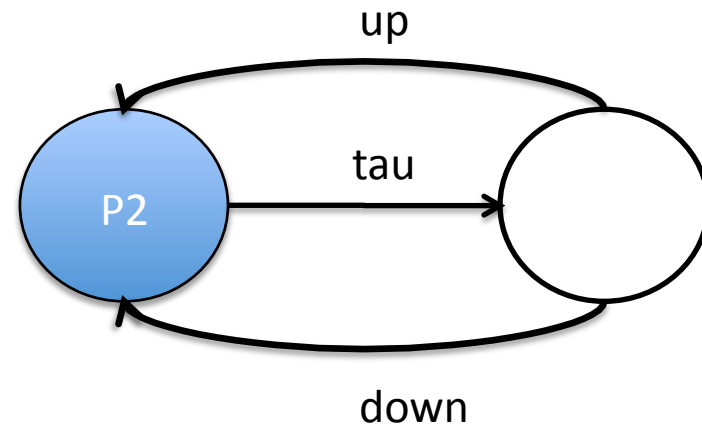
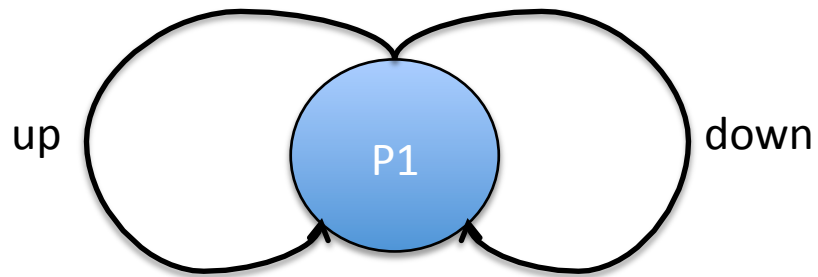
Exemplos de Comparação

- Q1 é igual a Q2?
 - Confirmar com FDR
- P4 é igual a P1?
 - Dica: leis algébricas $P \cap Q = Q \cap P$ e $P \cap P = P$
 - Confirmar com FDR

Exemplos de Refinamentos

- include “sample1.csp”
- Use FDR para responder as perguntas a seguir
- Igualdade de traces
 - $P1 =_T P2$?
 - $P2 =_T P3$?
- Refinamento de falhas
 - P1 e P2, quem tem menos falhas?
 - P2 e P3, quem tem menos falhas?

Exemplo de Refinamentos



Navegando
no Probe
chegamos a
estes LTSs

Exemplo Paralelismo Síncrono

- Especifique em CSPM o processo REPEAT e $\Sigma = \{\text{up}, \text{down}\}$
- Navegue com o Probe pelo processo
 - $\text{REPEAT}_{\text{up}} = (\text{up} \rightarrow \text{up} \rightarrow \text{REPEAT}) [|\Sigma|]$
REPEAT
- Seja
 - $\text{AS1} = \text{up} \rightarrow \text{up} \rightarrow \text{REPEAT}$
- Use FDR para responder
 - $\text{AS1} =_{\tau} \text{REPEAT}_{\text{up}} ?$

Exemplo Paralelismo Síncrono

- Especifique em CSPM o processo REPEAT e $\Sigma = \{\text{up}, \text{down}\}$
- Navegue com o Probe pelos processo
 - $\text{REPEAT}_{\text{updown}} = (\text{up} \rightarrow \text{down} \rightarrow \text{REPEAT}) [| \Sigma |] \text{REPEAT}$
- Seja
 - $\text{AS2} = \text{up} \rightarrow \text{STOP}$
- Use FDR para responder
 - $\text{AS2} =_{\top} \text{REPEAT}_{\text{updown}} ?$

Exemplo Paralelismo Alfabético

- Considere a seguinte especificação CSPM

$T = \{0,1\}$

channel $aa,m,cc:T$

$COPY'(a,b) = a?x \rightarrow b!x \rightarrow COPY'(a,b)$

$CC0 = COPY'(aa,m) [\{ |aa,m| \} \mid \{ |m,cc| \}]$

$COPY'(m,cc)$

- Escreva uma especificação sequencial equivalente a $CC0$