Design and Analysis of Distributed, Interacting Systems (DIS)

Tutorial 09
Raphael Pham
Raphael.Pham@inf.uni-hannover.de
Construction of a GBA

• ... is actually not that important – in practice, it is used differently.
• in oral exams, you need to know:
  – What is the general idea – why did we do this
  – What the construction idea looks like
  – How the states would look like (maximally consistent sets), what the idea is for the transitions (maintaining the LTL-attributes)
• This is a specification of how our coffee machine should behave
  • consisting of two MSDs.

\[ \Sigma = pB, bW, prC, poC, dC \]
\[ \Sigma_{sys} = ? \]
\[ \Sigma_{env} = ? \]

System Objects?
Environment Objects?
BA of MSDs

**MSD PrepareCoffee**

- **u:** User
- **c:** Controller
- **bu:** BrewerUnit

- **(c/m):** pressButton
- **(h/e):** boilWater
- **(h/e):** prepareCoffee
- **(h/e):** pourCoffee

Graph:

- **C0:**
  - **pB**
  - **bW**
  - **prC**

- **C1:**
  - **bW**

- **C2:**
  - **prC**

- **C3:**
  - **poC**

- **ERR:**
  - **Σ \ poC**

Symbols:

- **Σ \ pB**
- **Σ \ bW**
- **Σ \ prC**
- **Σ \ poC**

Raphael Pham, DIS-Tutorial 09
BA of MSDs

MSD DispenseCup

<table>
<thead>
<tr>
<th>c:Controller</th>
<th>bu:BrewerUnit</th>
<th>cd:CupDisp</th>
</tr>
</thead>
<tbody>
<tr>
<td>(c/m) prepareCoffee</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(c/m) pourCoffee</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
• State reactions of the play-out algorithm in response to pressButton
  – consider synchrony assumption
• pB, bW, prC, dC, pC
System Controller, Env Controller

- pB, bW, prC, dC, pC
Enhanced coffee machine

Play-Out run?

\( p_B, b_W, w_R, pr_C, d_C, c_R, po_C \)

\( p_B, b_W, p_B \) (cold violation), \( b_W, \ldots \)

\( p_B, b_W, w_R, pr_C, d_C, p_B \) (safety violation - stuck)

\( p_B, b_W, c_R, c_R, \ldots, w_R/p_B \)
System Controller

1. cold violation with first message
2. environment message no violation
3. cR
4. environment message - safety violation
5. poC
6. environment message - safety violation
7. pB

\[ \sum \]

Raphael Pham, DIS-Tutorial 09
Assumption MSD for pruning $C_{Sys}$

After $dC$, $cR$ will be true without interception

- **u:** User
- **c:** Controller
- **bu:** BrewerUnit
- **cd:** CupDisp

**Forbidden**
- `pushButton` -> `h`
- `waterReady` -> `h`

Disassembly:
- `dispenseCup` -> `(h/e)`
- `cupReady` -> `(c/m)`