

GOAL and Plugin Demo Script

Daniel Weibel

August 18, 2015

Preparation

- (Fresh) Mac version of GOAL (2014-11-17) installed
- Command line prepared by sourcing:

```
cd /Applications/GOAL.app/Contents/Resources/Java
history -s ./gc complement -m fribourg -r2c -m1 -m2 '~/Desktop/in.gff'
history -s ./gc complement -m piterman '~/Desktop/in.gff'
```

Demonstration of GOAL

- Start GOAL
- Generate random automaton ($dt = 2.4$, $da = 0.4$)
- Complement it
- Open property window by double-clicking on window
- Save input automaton to file ~/Desktop/in.gff
- In command line cat ~/Desktop/in.gff
- Complement automaton on command line
 - ./gc complement -m piterman ~/Desktop/in.gff (one step back in history)
 - ./gc complement -m piterman ~/Desktop/in.gff >~/Desktop/out_piterman.gff
- Switch to GUI and drag ~/Desktop/out_piterman.gff into it
- Make an equivalence test between the two output automata (Tab 1 and Tab 2)
- Close GOAL

Demonstration of the plugin

- Switch to Preview and click on link on previous slide
- Unzip ~/Desktop/ch.unifr.goal.complement by double-clicking it
- Show content of folder ch.unifr.goal.complement
- Open finder in /Applications
- Right-click GOAL.app > Show package contents, go to Contents > Resources > Java
- Move ch.unifr.goal.complement from Desktop to plugins
- Close Finder
- Open GOAL via icon in dock
- Drag ~/Desktop/in.gff into it
- Automaton > Complement > Fribourg construction
- Select R2C, M1, and M2 options
- Switch to command line
 - ./gc help complement (type)
 - ./gc complement -m fribourg -r2c -m1 -m2 ~/Desktop/in.gff (in history)
- Switch to GUI
- Possible to test equivalence between complements of Fribourg and Piterman, but takes too long
- Open Tab 0 (in.gff)
- Automaton > Complement (Step by Step) > Fribourg construction
 - Some steps manual (>|) then automatic (>>) (upper part)
 - Some steps manual (>|) then automatic (>>), and switch to another tab for some seconds
- Close GOAL