SPIN

An explicit state model checker
How does Spin work?

• We already saw:
  – The Algorithm
  – The Promela Language

• We need to see how we does the tool work.
The Buchi automaton is turned into a Promela process and composed with the rest of the system.

The generated verifier is specific to the model and property we started with.
Command Line Tools

• Spin
  – Generates the Promela code for the LTL formula
    ```
    ~$ spin –f “[]<>p”
    ```
  • The proposition in the formula must correspond to `#defines`
  – Generates the C source code
    ```
    ~$ spin –a source.pro
    ```
  • The property must be included in the source

• Pan
  – Performs the verification
    • Has many compile time options to enable different features
    • Optimized for performance
Xspin

- GUI for Spin
Simulator

- Spin can also be used as a simulator
  - Simulated the Promela program
- It is used as a simulator when a counterexample is generated
  - Steps through the trace
  - The trace itself is not “readable”
- Can be used for random and manually guided simulation as well
Comments

• DFS does not necessarily find the shortest counterexample
• There might be a very short counterexample but the verification might go out of memory
• If we don’t finish we might still have some sort of a result (coverage metrics)