Behavioral Verification

19CSE205 : PROGRAM REASONING

Dr. Swaminathan J

Assistant Professor

Department of Computer Science and Engineering

AMRITA VISHWA VIDYAPEETHAM

Jul - Dec 2020

Contents

Motivation

- Functional verification has its limits
  - It is input-output based.
  - Termination is necessary to prove correctness.

... verify behavior

- Applications are large and complex
  - Event driven programming
  - Concurrent/distributed systems
  - Multi-tiered architectures

- Reasoning with code is hard
  - Implementation contains unnecessary details.
  - Program code is not precise and unambiguous like math.

... reason with models

Move towards model-based behavioral verification!

Verifying code vs. model

Verification	Code	Model
Functional	ACSL & Frama-c JML & ESC/Java	Alt-ergo Why-3
Behavioral	BLAST Java PathFinder	SPIN TLA+

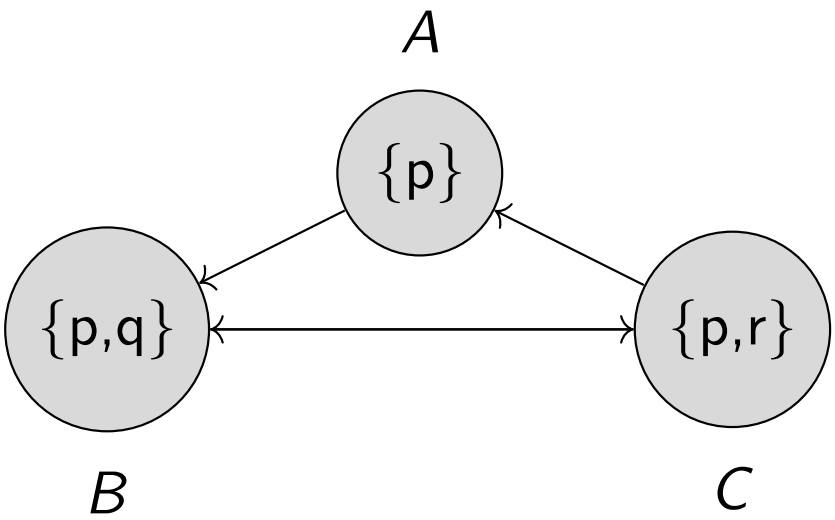
Broad goals in behavioral verification

1. Safety: Nothing bad ever happens.
2. Liveness: Something good will happen eventually.
3. Fariness: All get to progress.

Examples	Safety	Liveness	Fairness
Traffic light	Not two signals are green at same time	Not all signals are red at same time	No signal has to wait indefinitely to turn green
Counter	Incr & decr is not done at same time		
	Counter value does not turn negative		
Readers Writers	When a writer is active, no other process is active		Writer does not get blocked out by continuous readers
Dining Philosophers	No 2 adjacent phils eating at same time	Not all phils are hungry with one fork	No phil waits indefinitely hungry
Sleeping Barber	Only one customer serviced at a time	Barber not sleeping when customer waits	Customer serviced in the order of arrival
Critical Section	No 2 processes enters critical section at the same time	No 2 processes holds partial resources & awaits the other	No single process enters critical section without exiting for long

What does behavioral verification involve?

- A Model M of the system in the form of finite state machine.
- Behavior stated in the form of temporal property (p).
- A verification engine that verifies if M satisfies the property p. i.e. Does M $\models$ p? This process is called model checking.



Finite State Model

Property
Is p true in all states?
Is r true at any state?

Paths/Traces/Runs
Path 1: A B C A B C B C .....
Path 2: A B C B C B C A .....

Examples of Finite State Models 1/3

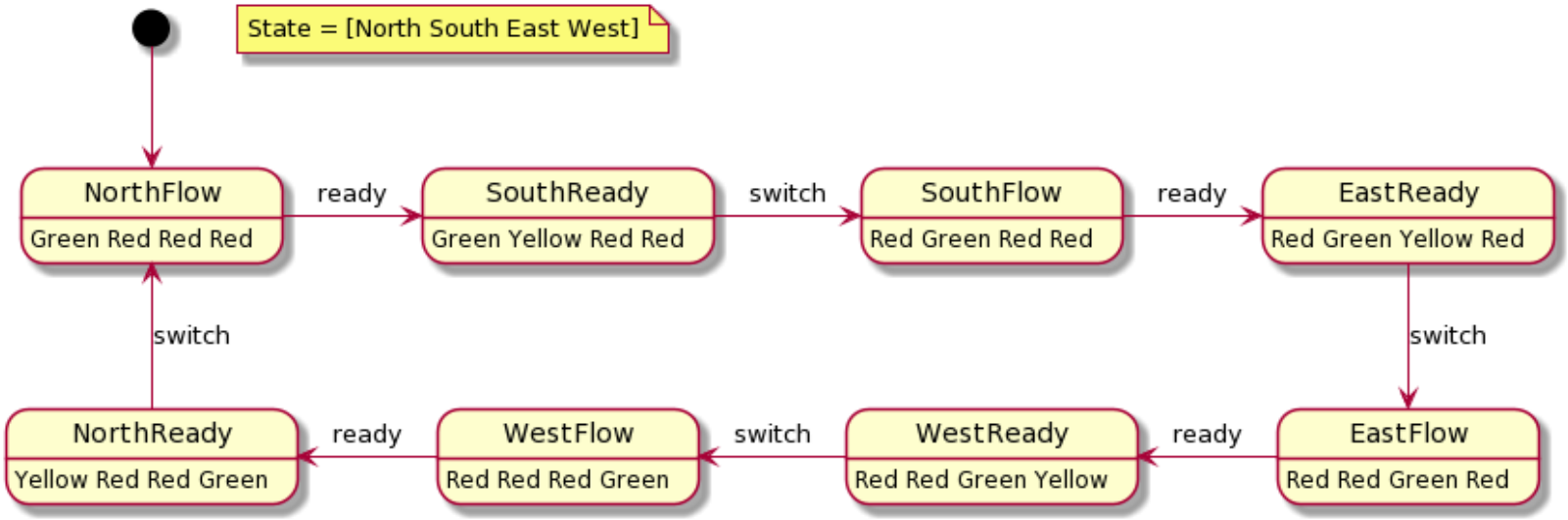


Figure: Finite State Model for Traffic Lights

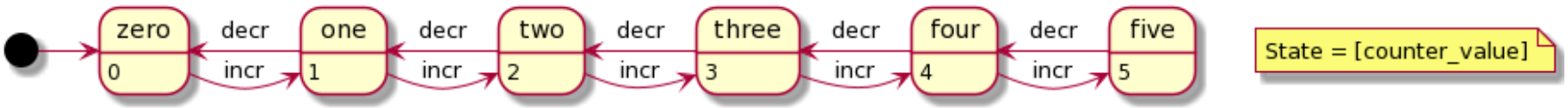


Figure: Finite State Model for Counter

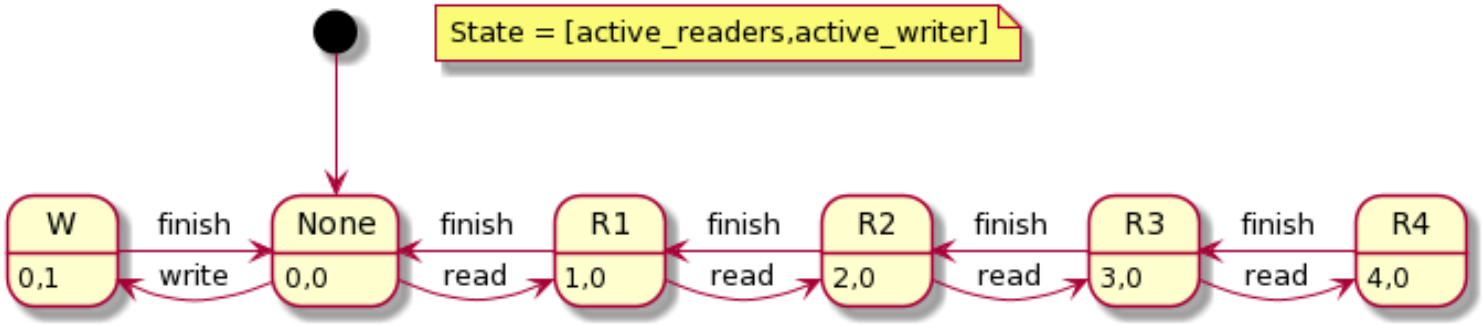


Figure: Finite State Model for Multiple Readers - Single Writer

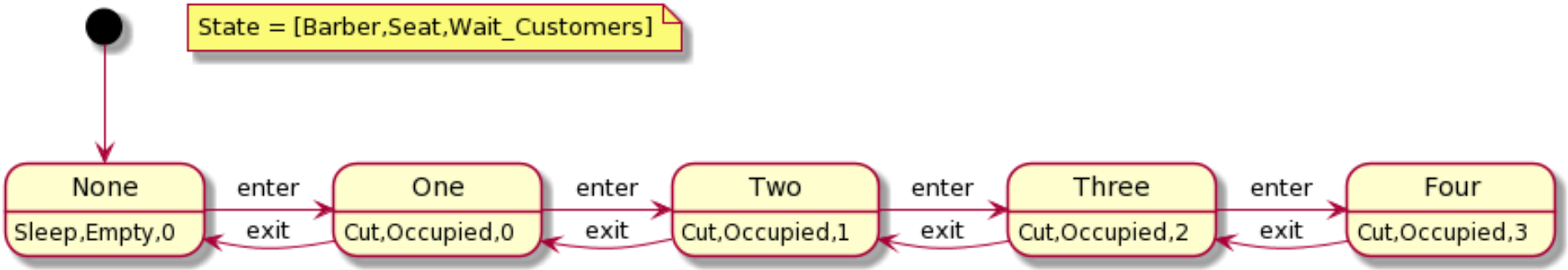


Figure: Finite State Model for Sleeping Barber

Examples of Finite State Models 3/3