Finite State and Buchi Automata Continued, CTL

Lecture Scribe: Rebecca Graber

March 29, 2016

Review

- Every model of LTL is a transition system, which can be represented by a directed graph / finite state automata (FSA)
- Every finite trace through an FSA can be represented as a regular language or expression and vice versa. The FSA does not have to be deterministic
- To use infinite traces, we must use Buchi automata. Every infinite trace through a Buchi automata can be represented as an $\omega$-regular language or expression and vice versa
- A note about determinism: deterministic Buchi automata define less than non-deterministic Buchi automata. The opposite is true for FSAs.

Composition of Transition Systems

- Sequential Composition: every final state in $A_1$ is connected to every start state in $A_2$

![Sequential Composition Diagram](image1)

Note that sequential composition is not symmetric

- Synchronous Composition ($A_1 \otimes A_2$): Move through both automata simultaneously

![Synchronous Composition Diagram](image2)
Asynchronous Composition ($A_1 || A_2$): Transitions are interleaved

Language of composed automata
- With $A_1, A_2$ as before: $L(A_1) = (ab)^*a = \{a, aba, ababa, \ldots\} = \{(ab)^n|n \geq 0\}$
- A Buchi automaton accepts an input iff it hits the acceptance states infinitely often
  $L_\omega(A_1) = (ab)^\omega = L_\omega(A_1 \otimes A_2)$
- To determine $L(A_1 || A_2)$:
  1. Define all finite sequences from 1A back to 1A without visiting 1A in between: $a(ab)^*b$
  2. Define all finite sequences from 1A back to 1A (no restriction): $(a(ab)^*b)^*$
  3. Define all finite sequences from 1A to 2B: $a(ab)^*a$
  4. Combine 1 and 3: $(a(ab)^*b)^*a(ab)^*a$

CTL
- CTL looks at all possibilities from one state
- syntax: $\varphi ::= <\text{propositional logic}>$
  $| AX\varphi | EX\varphi$
  $| AF\varphi | EF\varphi$
  $| AG\varphi | EF\varphi$
  $| A[\varphi U\varphi] | E[\varphi U\varphi]$
  $| A[\varphi R\varphi] | E[\varphi R\varphi]$
  $| A[\varphi W\varphi] | E[\varphi W\varphi]$
- CTL is also modeled by transition systems
- A WFF of CTL is defined with respect to a model $M$ and a state $s$
  For example: $M, s \models AX\varphi$ iff for all $s'$, $s \rightarrow s'$ in a path means $M, s' \models \varphi$
  $M, s \models EX\varphi$ iff there exists $s'$ such that $s \rightarrow s'$ in a path means $M, s' \models \varphi$
Visualization of semantics (defined with respect to tree root)

AGp

AFp

AXp

A[pUq]
Some English translations

- $\text{EF} \varphi \approx \text{potentially } \varphi$
- $\text{AF} \varphi \approx \text{inevitably } \varphi$
- $\text{EG} \varphi \approx \text{potentially always } \varphi$
- $\text{AG} \varphi \approx \text{invariantly } \varphi$