Chapter 4

Analysis

This chapter, yet to be written, gives an overview of specification formalisms and algorithmic techniques for the analysis of embedded systems.

4.1 Simulation

4.2 Temporal Logic

Temporal logic is a succinct mathematical notation to describe timing-related properties of systems. Using temporal logic, one can express properties about the occurrence of events, causal dependency between events, ordering of events, etc.

We will study a particular kind of temporal logic known as linear temporal logic, or LTL. An LTL formula expresses a property over a single trace of a system.

Recall that a trace of a finite-state machine is a sequence of the form

\[ s_0, s_1, s_2, s_3, \ldots \]

where \( s_j = (q_j, i_j, o_j) \) where \( q_j \) is the state at step \( j \), \( i_j \) is the input symbol at step \( j \), and \( o_j \) is the output symbol at step \( j \).

The simplest form of an LTL formula is an atomic formula. An atomic formula \( p \) is simply a function that maps \( s_j \) to \( \{0, 1\} \). If \( p(s_j) = 1 \), we say that \( p \) holds in \( s_j \) or \( s_j \) satisfies \( p \).

We say that atomic formula \( p \) holds for the trace \( s_0, s_1, s_2, \ldots \) if and only if \( p \) holds in \( s_0 \).

More complicated formulas can be constructed using the standard Boolean operators: AND (denoted \( \land \)), OR (denoted \( \lor \)), NOT (denoted \( \lnot \)) and IMPLIES (denoted \( \implies \)). The meaning of such a formula \( \phi \) is just the usual meaning for Boolean operators. For example, \( \phi_1 \land \phi_2 \) holds for a trace iff \( \phi_1 \) holds for the trace and \( \phi_2 \) also holds for the trace.
In addition, there are four special temporal operators:

- **G** which is read as “globally”. $G\phi$ holds for the trace if and only if, for all $j \geq 0$, formula $\phi$ holds in the suffix $s_j, s_{j+1}, s_{j+2}, \ldots$

- **F** which is read as “eventually”. $F\phi$ holds for the trace if and only if, for some $j \geq 0$, formula $\phi$ holds in the suffix $s_j, s_{j+1}, s_{j+2}, \ldots$

- **X** which is read as “next state”. $X\phi$ for the trace if and only if $\phi$ holds for the trace $s_1, s_2, s_3, \ldots$

- **U** which is read as “until”. $\phi_1 U \phi_2$ holds for the trace if and only if there exists $j \geq 0$ such that $\phi_2$ holds in the suffix $s_j, s_{j+1}, s_{j+2}, \ldots$ and $\phi_1$ holds in suffixes $s_i, s_{i+1}, s_{i+2}, \ldots$, for all $0 \leq i < j$.

  (Note that $\phi_1$ is not required to hold for $s_j, s_{j+1}, s_{j+2}, \ldots$)

Note that $\phi$, $\phi_1$, $\phi_2$ can themselves be temporal logic formulas. In other words, LTL formulas can be nested.

Note also that in the above discussion, we have evaluated LTL formulas over the entire trace, starting at step 0. It is also possible, and often useful, to evaluate an LTL formula at the suffix of a trace starting at step $j$.

For example, we can say that $X\phi$ holds for the trace $s_j, s_{j+1}, s_{j+2}, \ldots$ if and only if $\phi$ holds for the trace $s_{j+1}, s_{j+2}, \ldots$.

See the lecture slides for examples of LTL formulas.

### 4.3 Reachability Analysis

### 4.4 Model Checking