Abstract:
We provide an overview of graph-based verification patterns used in recent projects at IBM. We present a basic graph structure and show how to randomly traverse a finite state machine. We proceed to give an example of graph-based error injection. We extend this approach to end-to-end checking and error recovery. Finally, we show how to use reachability analysis to sanity-check testcases, improving the debuggability of complex scenarios.