one path: G: always, F: eventually, U: until, X: next

all paths: A, E

{ light: Green | Red
, power: running | broken } ⇒ | running × | Green
                              | broken | Red

F(¬p ∧ q)

p U_q ↯ G(p → X q)

AG(Green → AX(red ∨ power broken)

↯ AG((Green ∧ ¬power broken) → AX red)

E X φ	AG	EF
E φ1 U φ2	AF	
E G φ	AU	
	AX	

Strongly connected components (SCC)

partition the graph into SCC

Capture one infinite path

(• ∃ a cyclic path
that goes through all nodes
in the partition
• at least two nodes
if no self-loop allowed)