

# CIS 6930/4930 Computer Aided Verification

## Temporal Logics

Hao Zheng  
Dept. of Computer Science & Eng.  
Univ. of South Florida



USING LOGIC AND REASON,  
ERNEST AND WENDELL SETTLE A  
RULES INTERPRETATION DISPUTE  
AT WEDNESDAY NIGHTS GAME.

# Temporal Logics

- A formalism to describes properties of paths in the model of reactive systems.
- First order logic augmented with temporal operators.
- Time is implicit.
  - Explicit in real-time temporal logics.
- There exist different temporal logics.
  - With different view of underlying computation.
- CTL\* (CTL) views computation of a system as a tree.
  - System can move into different future.
- LTL views computation of a system as a set of paths.
  - System has only one direction into the future.



# Computational Tree Logic CTL\*

- To describe paths from a given state.
- Path quantifiers:
  - **A**: for all computation paths from a state.
  - **E**: for some computation path(s) from a state.
- Linear temporal operators: describe properties along a path.
  - **G** $p$  —  $p$  holds in every state on the path.
  - **F** $p$  —  $p$  holds in some state on the path.
  - **X** $p$  —  $p$  holds in the second state of the path
  - $p$ **U** $q$  —  $p$  holds until  $q$  holds in some state on the path.
  - $p$ **W** $q$  — similar to **U**, but  $q$  does not need to hold.

# State and Path Formulas

- Path formulas hold along a path.
  - If  $f$  is a state formula, it is also a path formula.
  - If  $f$  and  $g$  are path formulas, so are boolean combinations of  $f$  and  $g$ ,  $\mathbf{X}f$ ,  $\mathbf{F}f$ ,  $\mathbf{G}f$ , and  $f\mathbf{U}g$ .
- State formulas hold at a state.
  - If  $p$  is an atomic proposition, then  $p$  is a state formula.
  - If  $f$  and  $g$  are state formulas, so are boolean combinations of  $f$  and  $g$ .
  - If  $f$  is a path formula,  $\mathbf{A}f$  and  $\mathbf{E}f$  are state formulas.
- CTL\* formulae are state formulas generated by the above rules.

# Semantics: Path Formulas

- Defined w.r.t a Kripke structure  $M$ .
- If  $f$  is a path formula,  $M, \pi \models f$  means  $f$  holds along path  $\pi$ .
- Definitions:
  - $M, \pi \models f \leftrightarrow f$  is a state formula,  $s$  is the first state of  $M$ ,  $s \models f$  holds if  $p$  is an atomic proposition and  $p \in L(s)$ .
  - $M, \pi \models \neg f \leftrightarrow f$  is a path formula, and  $M, \pi \not\models f$  does not hold.
  - $M, \pi \models f \vee g \leftrightarrow f$  and  $g$  are path formulas, and  $M, s \models f$  **or**  $M, s \models g$ .
  - $M, \pi \models f \wedge g \leftrightarrow f$  and  $g$  are path formulas, and  $M, s \models f$  **and**  $M, s \models g$ .
  - $M, \pi \models \mathbf{X} f \leftrightarrow f$  is a path formula, and  $M, \pi^1 \models f$ .
  - $M, \pi \models \mathbf{F} f \leftrightarrow f$  is a path formula, and  $M, \pi^k \models f$  for some  $k \geq 0$ .
  - $M, \pi \models \mathbf{G} f \leftrightarrow f$  is a path formula, and  $M, \pi^k \models f$  for all  $k \geq 0$ .
  - $M, \pi \models f \mathbf{U} g \leftrightarrow \dots$

# Semantics: State Formulas

- Defined w.r.t a Kripke structure  $M$ .
- If  $f$  is a state formula,  $M, s \models f$  means  $f$  holds at state  $s$  of  $M$ .
- Definitions:
  - $M, s \models p \leftrightarrow$  if  $p$  is an atomic proposition and  $p \in L(s)$ .
  - $M, s \models \neg f \leftrightarrow M, s \not\models f$  does not hold.
  - $M, s \models f \vee g \leftrightarrow M, s \models f$  **or**  $M, s \models g$ .
  - $M, s \models f \wedge g \leftrightarrow M, s \models f$  **and**  $M, s \models g$ .
  - $M, s \models \mathbf{A} f \leftrightarrow f$  is a path formula, and for **all** paths  $\pi$  from  $s$  such that  $M, \pi \models f$ .
  - $M, s \models \mathbf{E} f \leftrightarrow f$  is a path formula, and there is **a** path  $\pi$  from  $s$  such that  $M, \pi \models f$ .

# Equivalences

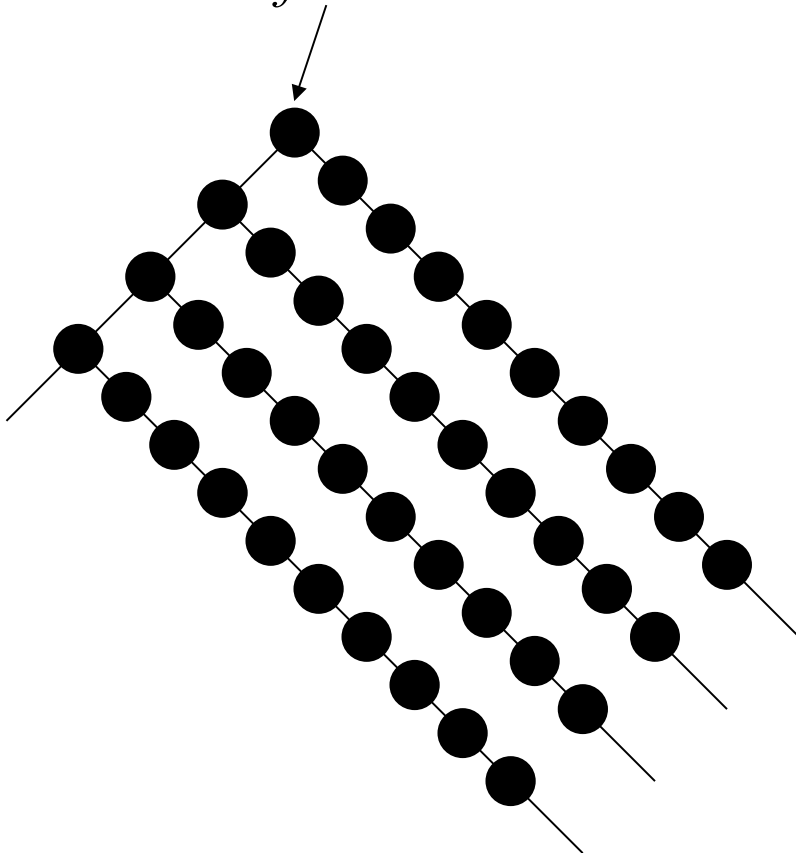
- Not all operators are essential to express a property.
  - $f \wedge g \equiv \neg(\neg f \vee \neg g)$
  - $\mathbf{A} f \equiv \neg \mathbf{E} (\neg f)$
  - $\mathbf{G} f \equiv \neg \mathbf{F} (\neg f)$
  - $\mathbf{F} f \equiv (\text{true} \mathbf{U} f)$
  - $\mathbf{F}(f \vee g) \equiv \mathbf{F}f \vee \mathbf{F}g$ 
    - What about  $\mathbf{F}(f \wedge g) \equiv \mathbf{F}f \wedge \mathbf{F}g$ ?
  - $\mathbf{G}(f \wedge g) \equiv \mathbf{G}f \wedge \mathbf{G}g$ 
    - What about  $\mathbf{G}(f \vee g) \equiv \mathbf{G}f \vee \mathbf{G}g$ .
  - $f \mathbf{U} g \equiv \neg(\neg g \mathbf{U} (\neg f \wedge \neg g)) \wedge \mathbf{F} g$

# CTL and LTL

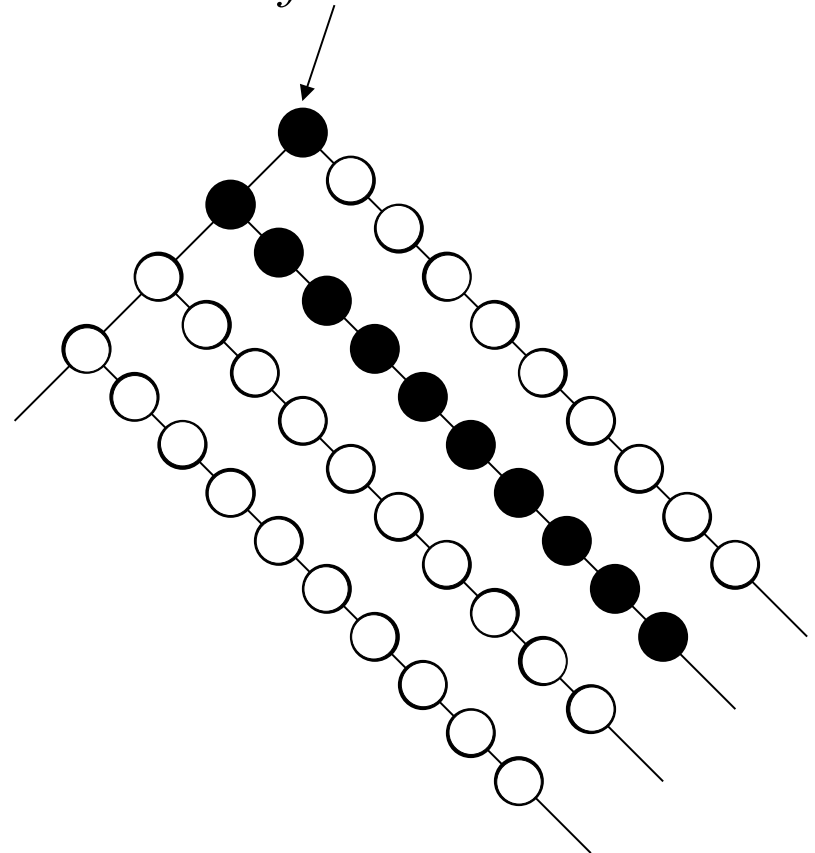
- CTL\* is more expressive, but expensive for verification.
- Two useful sublogics of CTL\*: CTL and LTL.
- CTL is a restricted subset of CTL\* where temporal operators must be immediately preceded by a path quantifier.
  - Basic operators: **AG**, **AF**, **AX**, **A(U)**, **EG**, **EF**, **EX**, **E(U)**.
  - Example: **AG( EF f )**
- LTL consists of formulas of the form **Af** defined as follows:
  - If  $p$  is an atomic formula, the  $p$  is a path formula.
  - if  $f$  and  $g$  are path formulas, so are boolean combinations of  $f$  and  $g$ , **Xf**, **Ff**, **Gf**, and  $f \mathbf{U} g$ .
  - Example: **A( FG f )**

# Interpretation of CTL Operators

**AG**  $f$  is true

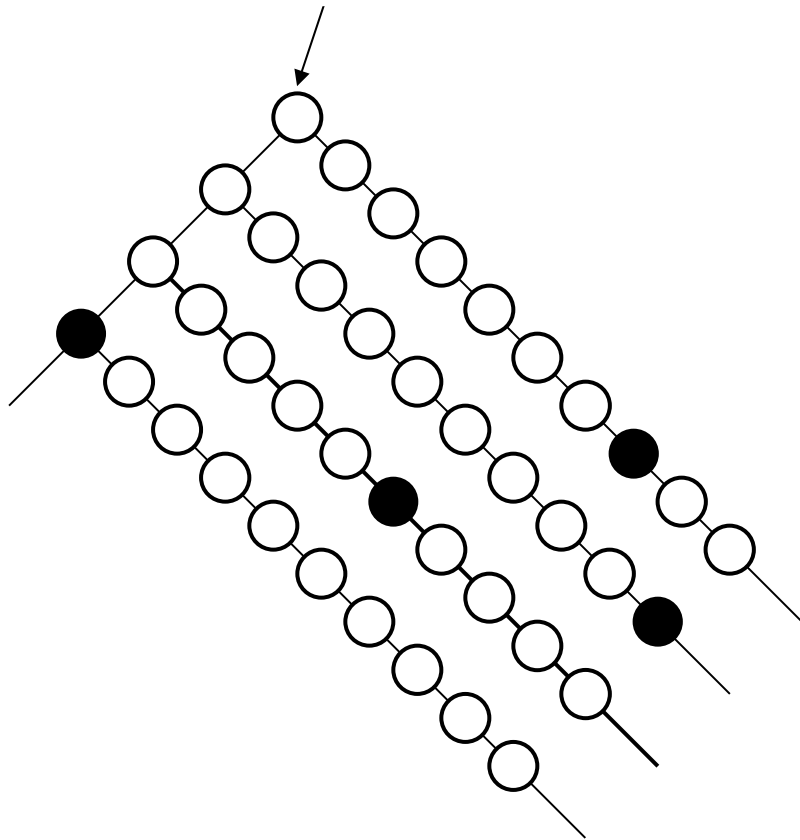


**EG**  $f$  is true

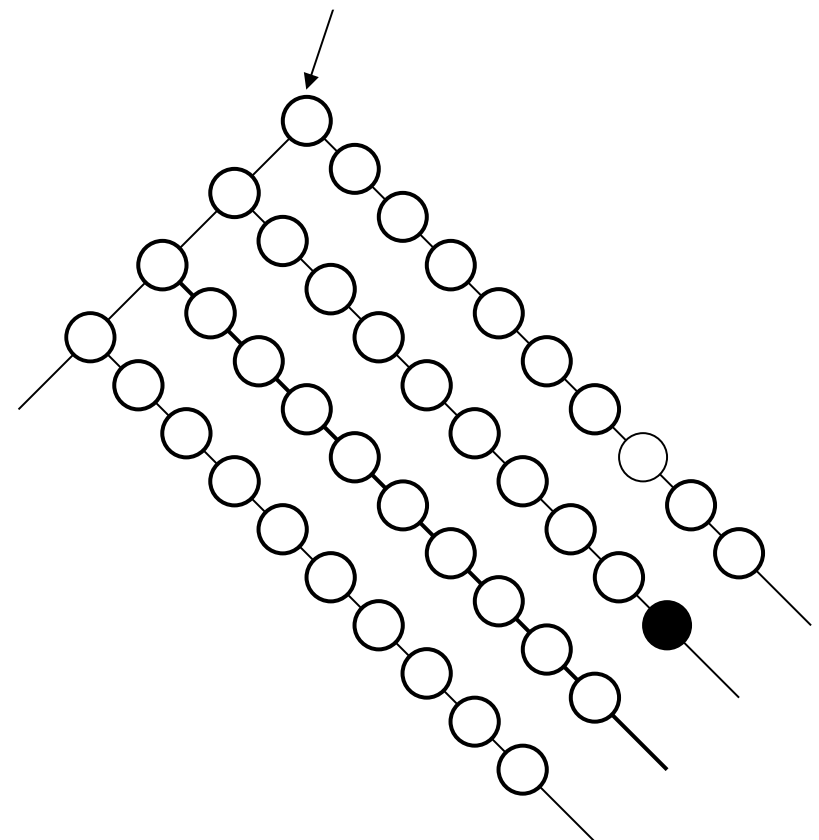


# Interpretation of CTL Operators

**AF**  $f$  is true

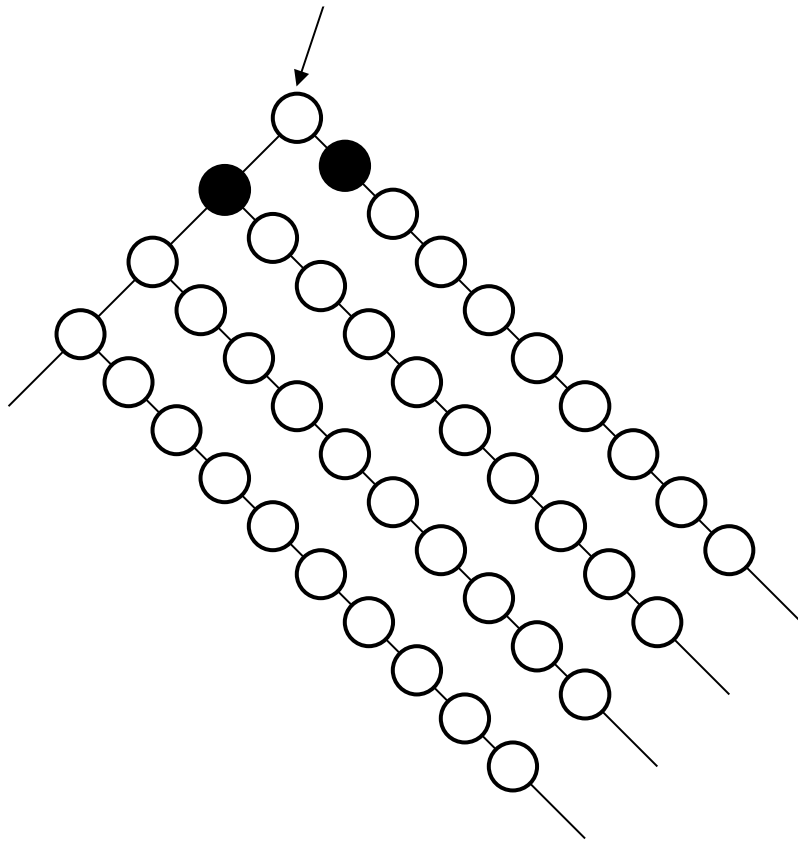


**EF**  $f$  is true

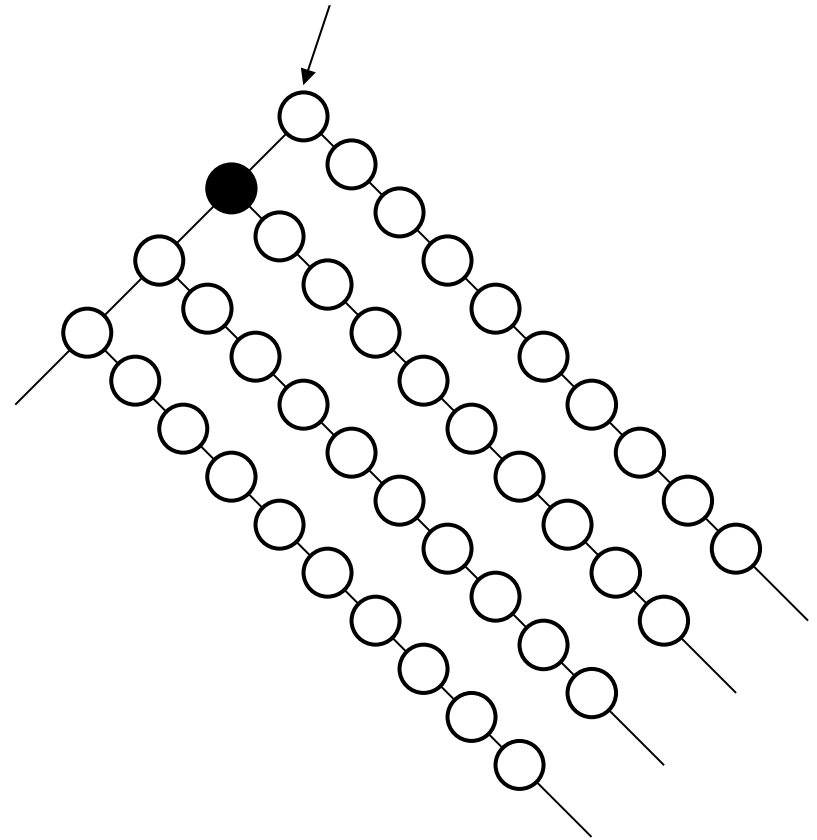


# Interpretation of CTL Operators

**AX**  $f$  is true



**EX**  $f$  is true

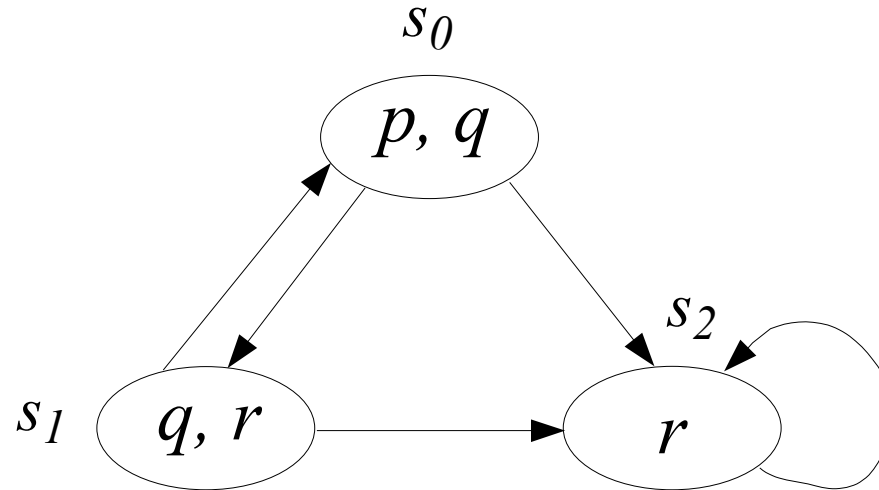




# A Sufficient Set of CTL Operators

- Any CTL formulas can be expressed using **EX**, **EG**, and **EU**.
  - $\mathbf{AX} f = \neg \mathbf{EX} \neg f$
  - $\mathbf{AG} f = \neg \mathbf{EF} \neg f = \neg \mathbf{E}(true \mathbf{U} \neg f)$
  - $\mathbf{AF} f = \neg \mathbf{EG} \neg f$
  - $\mathbf{A}(f \mathbf{U} g) = (\neg \mathbf{EG} \neg g) \wedge (\neg \mathbf{E}(\neg g \mathbf{U} (\neg f \wedge \neg g))) ?$
- What does  $\mathbf{AG}(\mathbf{AF} f)$  mean?

# LTL Semantics Example



$$M, s_0 \models p \wedge q$$

$$M, s_0 \models \mathbf{X} r$$

$$M, s_0 \models \mathbf{G} \neg(p \wedge r)$$

$$M, s_0 \models \mathbf{G} (\mathbf{F} p)$$

# A Sufficient Set of LTL Operators

- $\{\mathbf{U}, \mathbf{X}\}$ ,  $\{\mathbf{R}, \mathbf{X}\}$ , or  $\{\mathbf{W}, \mathbf{X}\}$  is sufficient.

- $\mathbf{G}f \equiv \neg \mathbf{F} \neg f$

- $\neg \mathbf{X}f \equiv \mathbf{X} \neg f$

- $f \mathbf{R} g \equiv \neg(\neg f \mathbf{U} \neg g)$

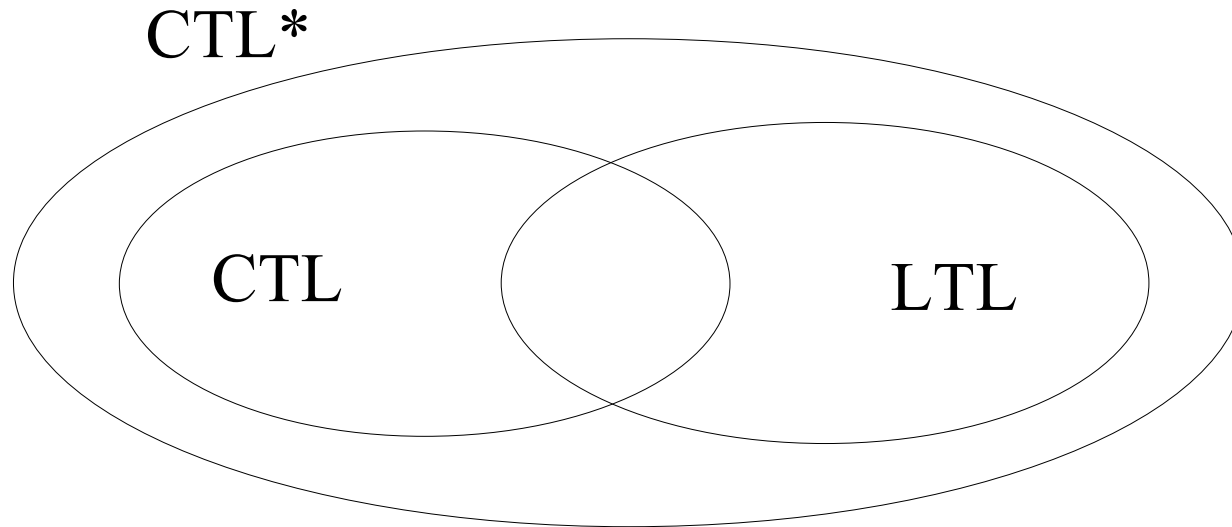
- $f \mathbf{U} g \equiv f \mathbf{W} g \wedge \mathbf{F}g$

- $\mathbf{F}f \equiv \mathbf{true} \mathbf{U} f$

- Examples:  $\mathbf{G}f$  and  $\mathbf{F}g$  ?

-

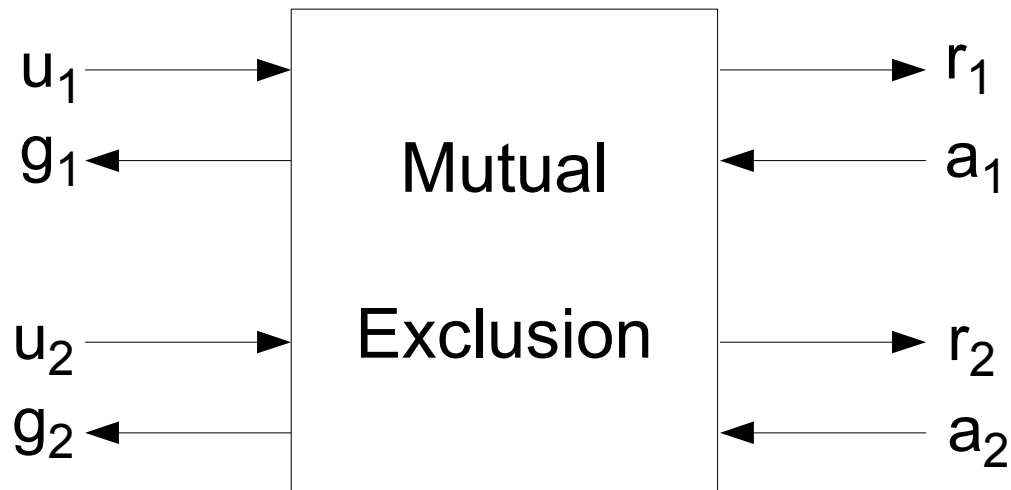
# CTL\*, CTL, and LTL



- CTL formula specifies a set of states.
- A LTL formula specifies a set of paths.
- **AFG**  $f$  is not expressible in CTL.
- **AG(EF**  $f$ ) is not expressible in LTL.

# Safety and Liveness Properties

- Safety: nothing bad should happen.
- Liveness: something good eventually happens.
- Example: a mutual exclusion element.



# Fairness

- Fairness means certain properties happen infinitely often during computation.
  - An arbiter cannot ignore some requests forever.
  - A communication channel cannot lose message all the time.
- Models may contain unfair computations.
  - Non-deterministic models of physical computing systems.
  - Wrong implementations of fairness requirements.
- Fairness constraints eliminate the unfair computations.
  - Unfairness introduced to simplify modeling.
- Fair computations satisfy fairness constraints infinitely often.

# Fair Semantics

- Fairness constraints are expressed as sets of states that hold infinitely often on computations.
- CTL\* semantics with fairness
  - $M, s \models_F p \leftrightarrow$  if there is a fair path from  $s$  and  $p \in L(s)$ .
  - $M, s \models_F \mathbf{A} f \leftrightarrow$   $f$  is a path formulas, and for **all fair** paths  $\pi$  from  $s$  such that  $M, \pi \models f$ .
  - $M, s \models_F \mathbf{E} f \leftrightarrow$   $f$  is a path formula, and there is **a fair** path  $\pi$  from  $s$  such that  $M, \pi \models f$ .
- CTL – will discuss it later.
- LTL – fairness can be easily expressed and incorporated with verification.
  - Ex.:  $\mathbf{GF} p$ , or  $\mathbf{GF} p \rightarrow \mathbf{GF} q$