Linear Temporal Logic (LTL)

Slides by Alessandro Artale
http://www.inf.unibz.it/∼artale/

Some material (text, figures) displayed in these slides is courtesy of: M. Benerecetti, A. Cimatti, M. Fisher, F. Giunchiglia, M. Pistore, M. Roveri, R. Sebastiani.

An Introduction to Temporal Logics

In classical logic, formulae are evaluated within a single fixed world.

For example, a proposition such as "it is Monday" must be either *true* or *false*.

Propositions are then combined using constructs such as '$\wedge$', '$\neg$', etc.

But, most (not just computational) systems are dynamic.

In temporal logics, evaluation takes place within a set of worlds. Thus, "it is Monday" may be satisfied in some worlds, but not in others.

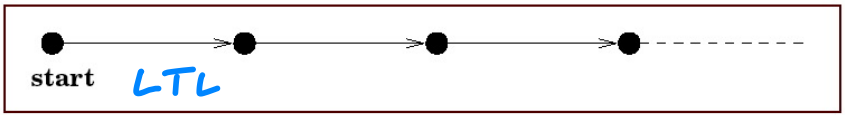
An Introduction to Temporal Logics (Cont.)

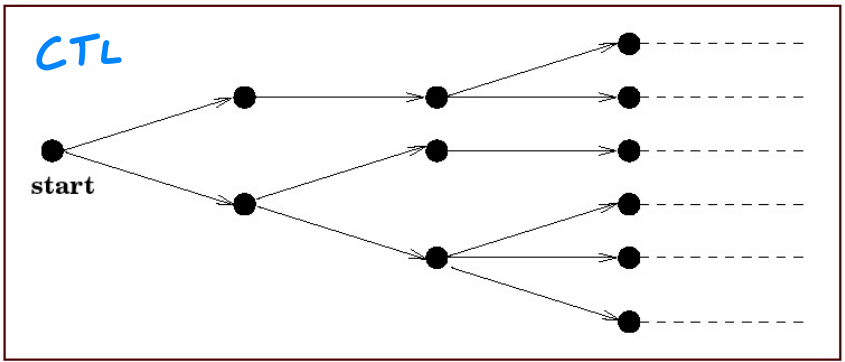
The set of worlds correspond to moments in time.
How we navigate between these worlds depends on our particular view of time.

The particular model of time is captured by a temporal accessibility relation between worlds.

Essentially, temporal logic extends classical propositional logic with a set of temporal operators that navigate between worlds using this accessibility relation.

Typical Models of Time





Linear Temporal Logic (LTL): Intuitions

Consider a simple temporal logic (LTL) where the accessibility relation characterises a discrete, linear model isomorphic to the Natural Numbers.

Typical temporal operators used are

NEXT	$\bigcirc\varphi$	φ is true in the *next* moment in time
ALWAYS	$\Box\varphi$	φ is true in *all* future moments
EVENTUALLY	$\Diamond\varphi$	φ is true in *some* future moment
UNTIL	$\varphi \mathsf{U} \psi$	φ is true *until* ψ is true

Example:

$$\Box((\neg passport \vee \neg ticket) \rightarrow \bigcirc \neg board\_flight)$$

Computational Example

- $\Box(\mathit{requested} \rightarrow \Diamond\, \mathit{received})$
- $\Box(\mathit{received} \rightarrow \bigcirc\, \mathit{processed})$
- $\Box(\mathit{processed} \rightarrow \Diamond\Box\, \mathit{done})$

From the above we should be able to infer that it is *not* the case that the system continually re-sends a request, but never sees it completed ($\Box\neg \mathit{done}$); i.e. the statement

$$\Box\, \mathit{requested} \wedge \Box\neg \mathit{done}$$

should be inconsistent.

LTL: Syntax

Countable set Σ of *atomic propositions*: $p, q, \ldots$ the set Form of formulas is:

$$
\begin{array}{lll}
\varphi, \psi \quad \rightarrow \quad & p \mid & \text{(atomic proposition)} \\
& \top \mid & \text{(true)} \\
& \bot \mid & \text{(false)} \\
& \neg \varphi \mid & \text{(complement)} \\
& \varphi \wedge \psi \mid & \text{(conjunction)} \\
& \varphi \vee \psi \mid & \text{(disjunction)} \\
& \bigcirc \varphi \mid & \text{(next time)} \\
& \Box \varphi \mid & \text{(always)} \\
& \Diamond \varphi \mid & \text{(sometime)} \\
& \varphi \, \mathsf{U} \, \psi & \text{(until)}
\end{array}
$$

Temporal Semantics

We interpret our temporal formulae in a discrete, linear model of time. Formally, this structure is represented by

$$\mathscr{M} = \langle \mathbb{N}, I \rangle$$

where

- $I : \mathbb{N} \mapsto 2^{\Sigma}$ maps each Natural number (representing a moment in time) to a set of propositions.

The semantics of a temporal formula is the satisfaction relation:

$$\models : (\mathscr{M} \times \mathbb{N} \times \mathsf{Form}) \to \{\mathsf{true}, \mathsf{false}\}$$

Semantics: The Propositional Aspect

We start by defining when an atomic proposition is true at a time point i:

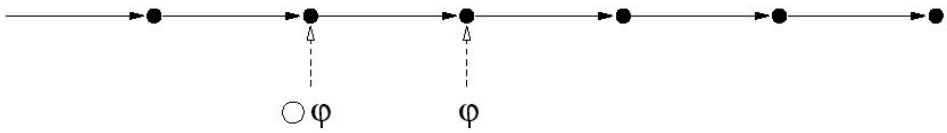
$$\langle \mathscr{M}, i \rangle \models p \text{ iff } p \in \mathscr{I}(i) \qquad (p \in \Sigma)$$

$$\begin{array}{ll}
\langle \mathscr{M}, i \rangle \models \neg\varphi & \text{iff } \langle \mathscr{M}, i \rangle \not\models \varphi \\
\langle \mathscr{M}, i \rangle \models \varphi \wedge \psi & \text{iff } \langle \mathscr{M}, i \rangle \models \varphi \text{ and } \langle \mathscr{M}, i \rangle \models \psi \\
\langle \mathscr{M}, i \rangle \models \varphi \vee \psi & \text{iff } \langle \mathscr{M}, i \rangle \models \varphi \text{ or } \langle \mathscr{M}, i \rangle \models \psi \\
\langle \mathscr{M}, i \rangle \models \varphi \rightarrow \psi & \text{iff } \langle \mathscr{M}, i \rangle \models \varphi \text{ implies } \langle \mathscr{M}, i \rangle \models \psi \\
\mathscr{M}, i \models \top & \\
\mathscr{M}, i \not\models \bot &
\end{array}$$

Temporal Operators: 'next'

$$\langle \mathscr{M}, i \rangle \models \bigcirc \varphi \text{ iff } \langle \mathscr{M}, i+1 \rangle \models \varphi$$

This operator constrains the next moment in time.

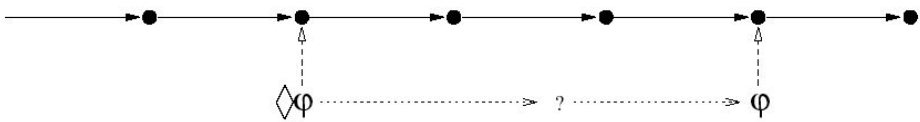


$$(\text{sad} \wedge \neg \text{rich}) \rightarrow \bigcirc \text{sad}, \qquad ((x = 0) \wedge \mathit{add}3) \rightarrow \bigcirc (x = 3)$$

Temporal Operators: 'sometime'

$$\langle \mathscr{M}, i \rangle \models \Diamond \varphi \text{ iff } \exists j \ (j \geq i) \ \langle \mathscr{M}, j \rangle \models \varphi$$

We know φ will be true now or later, but not when.

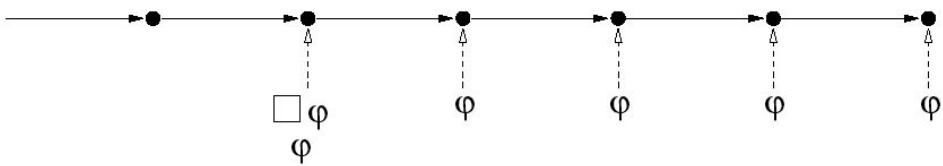


$(\neg \text{resigned} \wedge \text{sad}) \rightarrow \Diamond \text{famous}, \quad \text{send} \rightarrow \Diamond \text{receive}$

Temporal Operators: 'always'

$$\langle \mathscr{M}, i \rangle \models \Box \varphi \text{ iff } \forall j.\ (j \geq i) \Rightarrow \langle \mathscr{M}, j \rangle \models \varphi$$

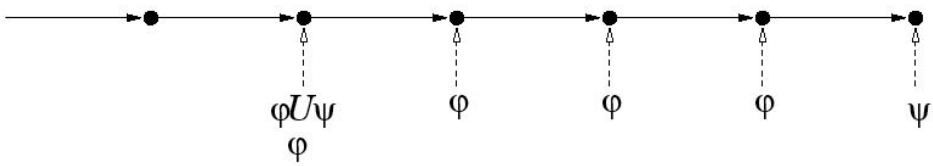
Represents invariants.



lottery-win $\rightarrow \Box$ rich

Temporal Operators: 'until'

$$\langle \mathscr{M}, i \rangle \models \varphi \,\mathsf{U}\, \psi \text{ iff } \exists j\ (j \geq i)\ \langle \mathscr{M}, j \rangle \models \psi\ \wedge \forall k.\ (i \leq k < j) \Rightarrow \langle \mathscr{M}, k \rangle \models \varphi$$



start_lecture $\rightarrow$ (talk until end_lecture), request $\rightarrow$ (reply until acknowledgement)

Satisfiability and Validity

A structure $\mathscr{M} = \langle \mathbb{N}, I \rangle$ is a model of ϕ, if

$$\langle \mathscr{M}, i \rangle \models \phi \quad \text{for some } i \in \mathbb{N}.$$

Similarly as in classical logic, an LTL formula ϕ can be:

- **Satisfiable**: there is a model for ϕ.
- **Unsatisfiable**: no model exists.
- **Valid (Tautology)**: $\models \phi$ iff $\forall \mathscr{M}, \forall i \in \mathbb{N}. \langle \mathscr{M}, i \rangle \models \phi$.

Entailment and Equivalence

Similarly as in classical logic, we can define the notions of **entailment** and **equivalence** between two LTL formulas

- **Entailment.** $\phi \models \psi$ iff $\forall \mathscr{M}, \forall i.\ \langle \mathscr{M}, i \rangle \models \phi \Rightarrow \langle \mathscr{M}, i \rangle \models \psi$.
- **Equivalence.** $\phi \equiv \psi$ iff $\forall \mathscr{M}, \forall i.\ \langle \mathscr{M}, i \rangle \models \phi \iff \langle \mathscr{M}, i \rangle \models \psi$.

Equivalences in LTL

The temporal operators $\Box$ and $\Diamond$ are duals:

$$\neg \Box \varphi \equiv \Diamond \neg \varphi$$

$\Diamond$ (and then $\Box$) can be rewritten in terms of U:

$$\Diamond \varphi \equiv \top \mathsf{U}\, \varphi$$

All temporal operators can be rewritten using “Until” and “Next”.

Equivalences in LTL (Cont.)

$\Diamond$ distributes over $\vee$ while $\Box$ distributes over $\wedge$:

$$\Diamond(\varphi \vee \psi) \equiv \Diamond \varphi \vee \Diamond \psi$$

$$\Box(\varphi \wedge \psi) \equiv \Box \varphi \wedge \Box \psi$$

The following equivalences are useful for generating formulas in Negated Normal Form:

$$\neg \bigcirc \varphi \equiv \bigcirc \neg \varphi$$

$$\neg(\varphi \, \mathsf{U} \, \psi) \equiv (\neg \psi \, \mathsf{U} \, (\neg \varphi \wedge \neg \psi)) \vee \Box \neg \psi$$

LTL Vs. FOL

LTL can be thought of as a specific decidable (PSPACE-complete) fragment of *classical first-order logic over linear infinite structures*

We just map each proposition to a unary predicate in FOL. In general, the following satisfiability preserving mapping holds:

$$
\begin{array}{lcl}
p & \rightsquigarrow & p(t) \\
\bigcirc p & \rightsquigarrow & p(t+1) \\
\Diamond p & \rightsquigarrow & \exists t'.\ (t' \geq t) \wedge p(t') \\
\Box p & \rightsquigarrow & \forall t'.\ (t' \geq t) \Rightarrow p(t')
\end{array}
$$

Temporal Logic in Computer Science

Temporal logic was originally developed in order to represent tense in natural language.

Within Computer Science, it has achieved a significant role in the formal specification and verification of concurrent reactive systems.

Much of this popularity has been achieved as a number of useful concepts can be formally, and concisely, specified using temporal logics, e.g.:

- *safety properties*
- *liveness properties*
- *fairness properties*

Safety Properties

Safety: "something bad will not happen"

$$\Box\neg(\mathit{reactor\_temp} > 1000)$$

$$\Box\neg((x = 0) \wedge \bigcirc\bigcirc\bigcirc(y = z/x))$$

Usually written as $\Box\neg\ldots$

Liveness Properties

Liveness: "something good will happen"

$$\Diamond\, rich, \quad \Diamond(x > 5), \quad \Box(start \rightarrow \Diamond\, terminate)$$

Usually written with $\Diamond \ldots$

Fairness Properties

Strong Fairness:

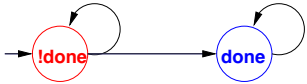
"if something is attempted/requested infinitely often, then it will be successful/allocated infinitely often"

$$\Box \Diamond \mathit{ready} \rightarrow \Box \Diamond \mathit{run}$$

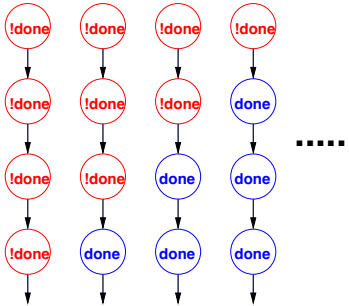
Often only really useful when scheduling processes, responding to messages, etc.

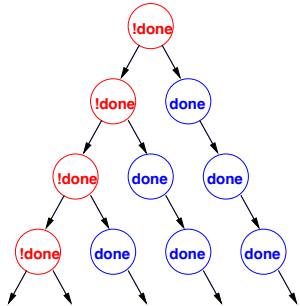
Transition systems and Linear Structures

Consider the following transition system:



Its paths/computations can be seen as a set of linear structures (computation tree):





Path-Semantics for LTL

- LTL formulae are evaluated over natural numbers $\mathbb{N}$.
- Paths in transition systems are infinite sequences of states: $\pi = s_0 \to s_1 \to \cdots$
- We want to interpret LTL formulas over transition systems.
- Given a transition system $\mathscr{T}$, a path π, a state s, and formula ϕ, we define $\langle \mathscr{T}, \pi \rangle \models \phi$ and $\langle \mathscr{T}, s \rangle \models \phi$ from the LTL semantics over $\mathbb{N}$.

Path-Semantics for LTL (Cont.)

Extract an LTL model $\mathscr{M}_\pi = (\pi, l_\pi)$ from $\mathscr{T}$:

- π is a path in $\mathscr{T}$.
- l_π is the sequence formed by propositional atoms labeling each state of π (i.e. $p \in l_\pi(s)$ iff $p \in L(s)$).

Then:

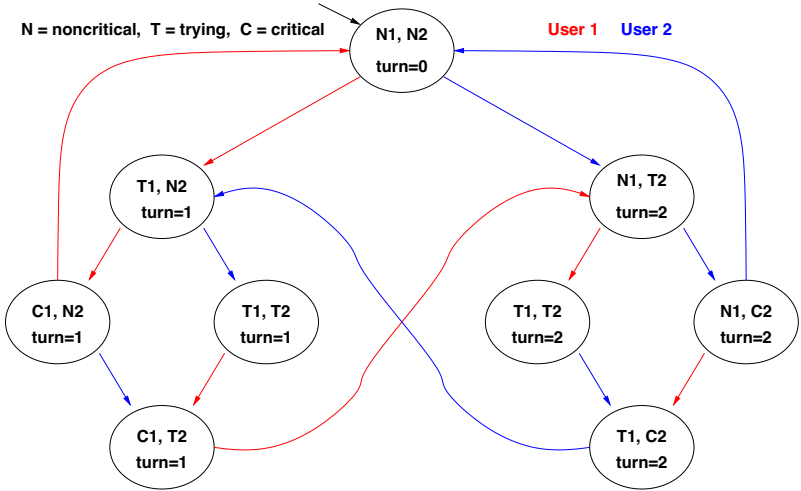
- $\langle \mathscr{T}, \pi \rangle \models \phi$ iff $\langle \mathscr{M}_\pi, s_0 \rangle \models \phi$.
- $\langle \mathscr{T}, s \rangle \models \phi$ iff $\langle \mathscr{T}, \pi \rangle \models \phi$ for all paths π starting at s.

LTL Model Checking Definition

Given $\mathscr{T}$, the LTL model checking problem $\mathscr{T} \models \phi$:

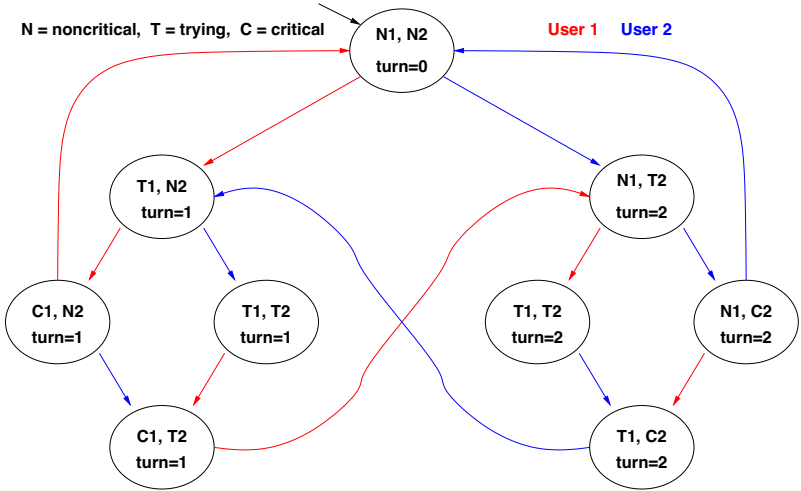
Check if $\langle \mathscr{T}, s_0 \rangle \models \phi$ ***for the/every initial state of*** s_0 ***of*** $\mathscr{T}$***.***

Example 1: mutual exclusion (safety)



$\mathscr{T} \models \Box\neg(C_1 \wedge C_2)$?

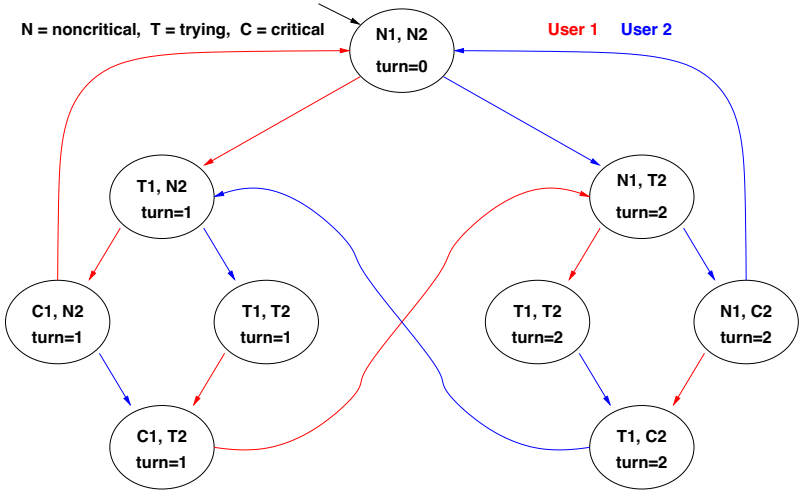
Example 1: mutual exclusion (safety)



$\mathscr{T} \models \Box\neg(C_1 \wedge C_2)$?

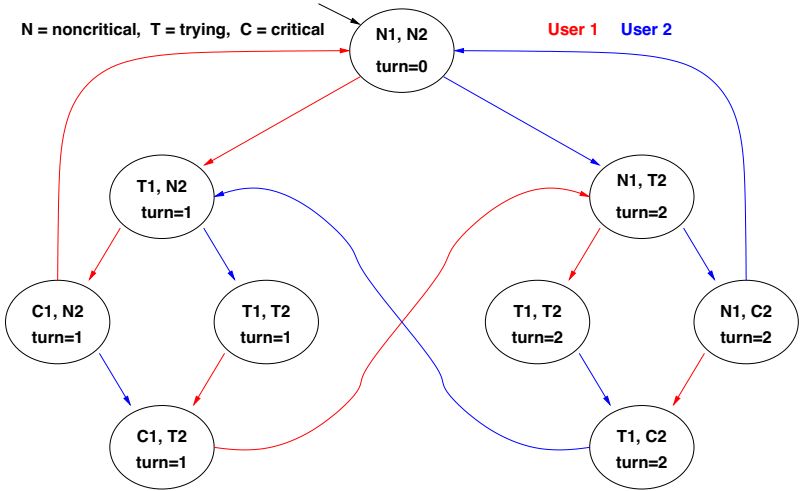
YES: There is no reachable state in which $(C_1 \wedge C_2)$ holds!

Example 2: mutual exclusion (liveness)



$\mathscr{T} \models \Diamond C_1$?

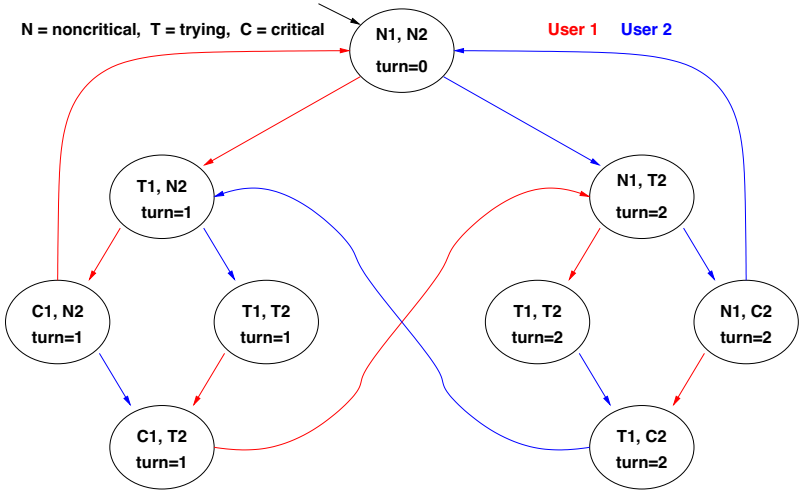
Example 2: mutual exclusion (liveness)



$\mathscr{T} \models \Diamond C_1$?

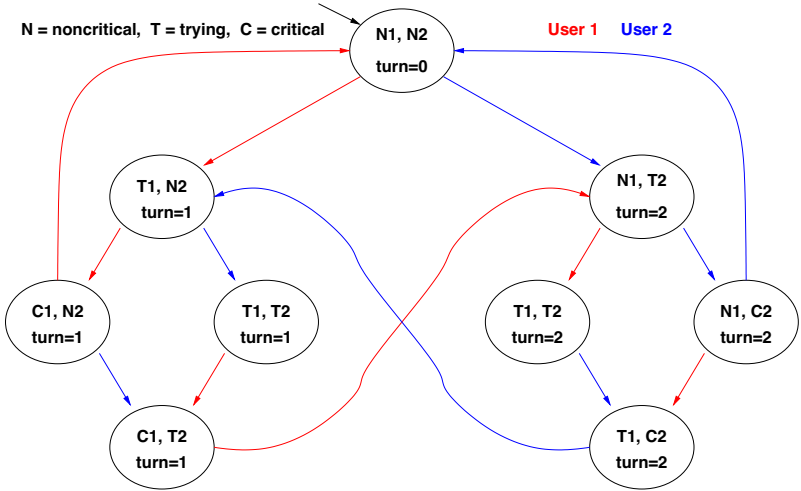
NO: the blue cyclic path is a counterexample!

Example 3: mutual exclusion (liveness)



$\mathscr{T} \models \Box(T_1 \to \Diamond C_1)$?

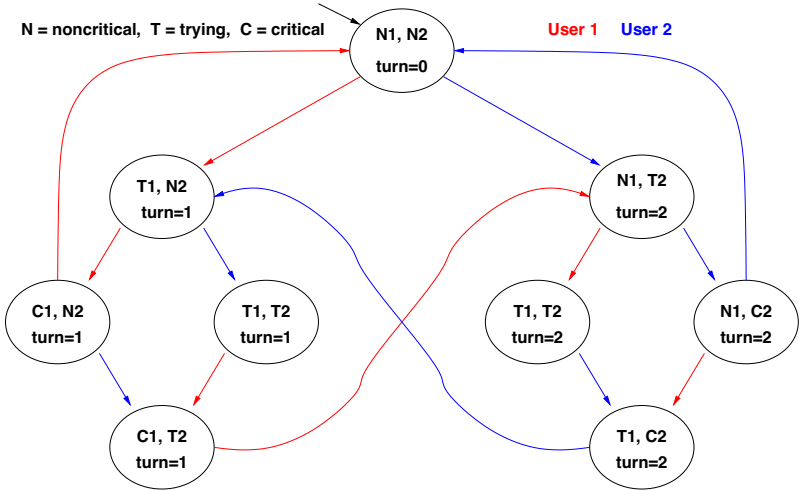
Example 3: mutual exclusion (liveness)



$\mathscr{T} \models \Box(T_1 \rightarrow \Diamond C_1)$?

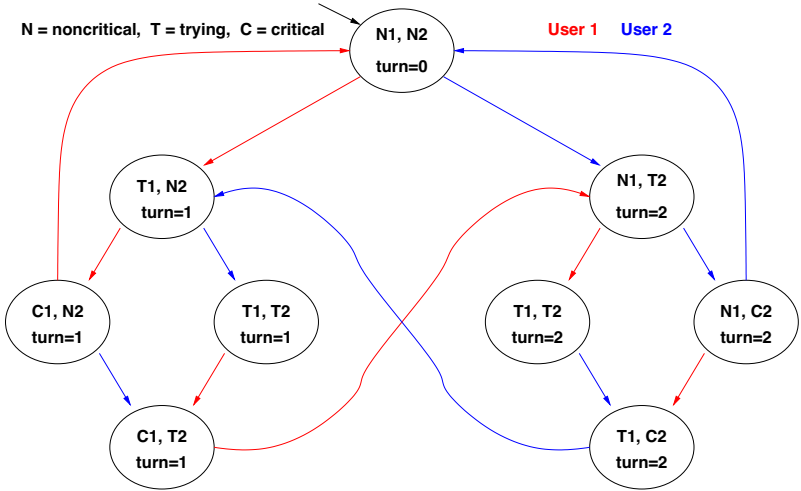
YES: in every path if T_1 holds afterwards C_1 holds!

Example 4: mutual exclusion (fairness)



$\mathscr{T} \models \Box \Diamond C_1$?

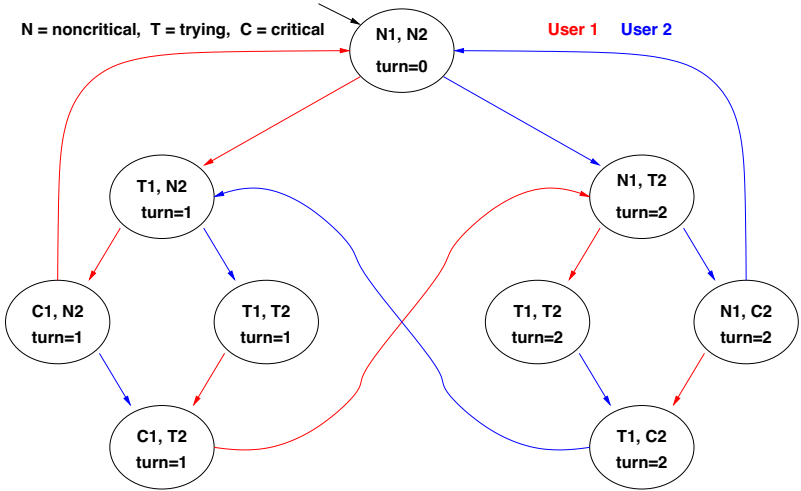
Example 4: mutual exclusion (fairness)



$\mathscr{T} \models \Box \Diamond C_1$?

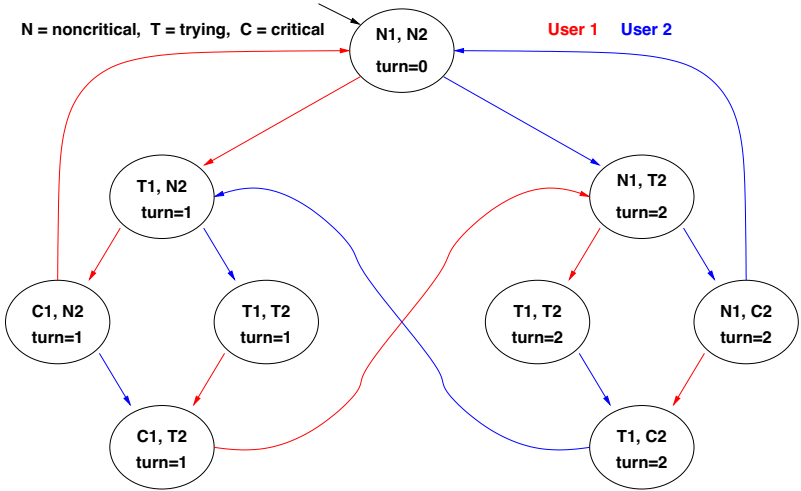
NO: the blue cyclic path is a counterexample!

Example 4: mutual exclusion (strong fairness)



$\mathscr{T} \models \Box\Diamond T_1 \rightarrow \Box\Diamond C_1$?

Example 4: mutual exclusion (strong fairness)



$\mathscr{T} \models \Box\Diamond T_1 \rightarrow \Box\Diamond C_1$?

YES: every path which visits T_1 infinitely often also visits C_1 infinitely often!

LTL Alternative Notation

Alternative notations are used for temporal operators:

$\Diamond$	$\rightsquigarrow$	F	sometime in the Future
$\Box$	$\rightsquigarrow$	G	Globally in the future
$\bigcirc$	$\rightsquigarrow$	X	neXtime