# CTL MODEL CHECKING

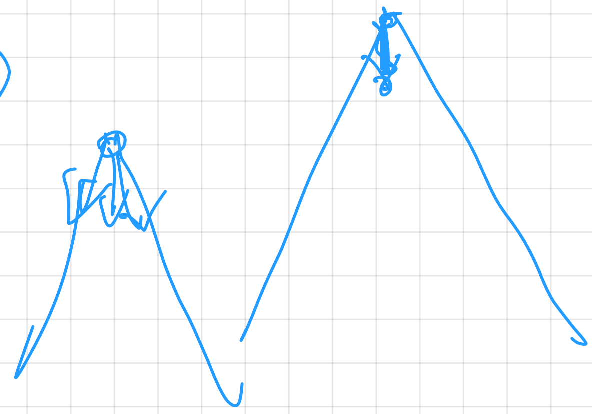
Slides by Alessandro Artale
`http://www.inf.unibz.it/~artale/`

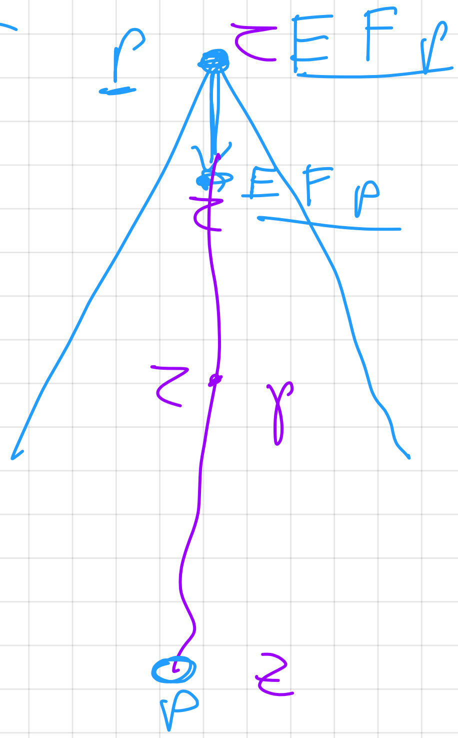
*Some material (text, figures) displayed in these slides is courtesy of:*
*M. Benerecetti, A. Cimatti, M. Fisher, F. Giunchiglia, M. Pistore, M. Roveri, R.Sebastiani.*

CTL $\qquad\qquad$ $\mu$-calc

$$t(EX\,\varphi) \rightsquigarrow \langle next \rangle\, t(\varphi)$$

$$t(AX\,\varphi) \rightsquigarrow [next]\, t(\varphi)$$





$$\boxed{EF\,p} \equiv p \lor EX\,\boxed{EF\,p}$$

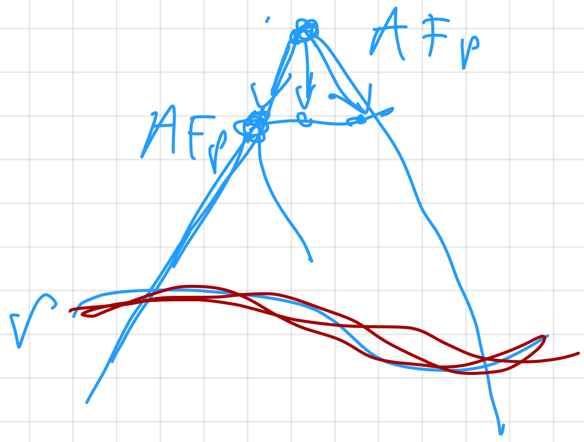
$$Z \equiv p \lor EX\,Z$$

lfp $\quad Z \equiv p \lor \langle next \rangle Z$

$$\mu Z.\ p \lor \langle next \rangle Z$$

$$t(EF\,\varphi) \longrightarrow \mu Z\ t(\varphi) \lor \langle next \rangle Z$$

$$\boxed{A\,F\,p} \equiv p \vee A\,X\,\boxed{A\,F\,p}$$




$$Z \equiv p \vee A\,X\,Z$$

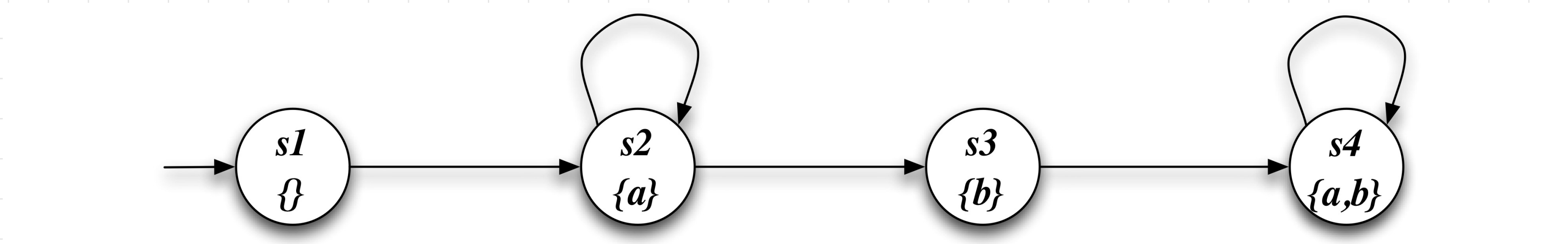
$$\mathrm{lfp}\ Z \equiv p \vee [-\mathrm{exr}]\,Z$$

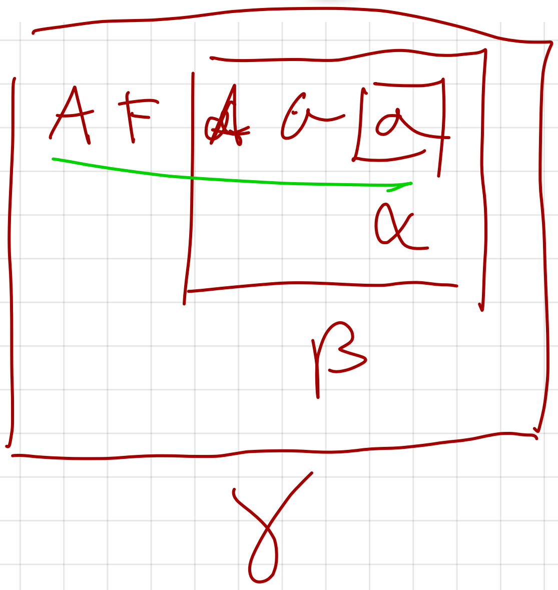
$$\mu Z.\ p \vee [-\mathrm{exr}]\,Z$$

$$T(A\,F\,\varphi) \rightsquigarrow \mu Z.\ T(\varphi) \vee [-\mathrm{exr}]\,Z$$

$TS, s1 \models AF\,AG\,a$ NO

$1 \notin \{3,4\}$




$[\![\alpha]\!] = [\![a]\!] = \{2,4\}$

$[\![\beta]\!] = [\![AG\,\alpha]\!] = \{4\}$

$= [\![\nu Z.\ \alpha \wedge [next]\, Z]\!]$

$[\![Z_0]\!] = \{1,2,3,4\}$

$[\![Z_1]\!] = [\![\alpha \wedge [next]\, Z_0]\!] =$

$= [\![\alpha]\!] \cap PreA(next, [\![Z_0]\!])$

$\{2,4\} \cap \{1,2,3,4\} = \{2,4\}$

$[\![Z_2]\!] = [\![\alpha \wedge [next]\, Z_1]\!]$

$\{2,4\} \cap \{1,3,4\} = \{4\}$

$[\![Z_3]\!] = [\![\alpha \wedge [next]\, Z_2]\!]$

$= [\![\alpha]\!] \cap PreA(next, Z_2)$

$\{2,4\} \cap \{3,4\} = \{4\}$

$\alpha = a$

$\beta = AG\,\alpha$

$\gamma = AF\,\beta$

$[\![\gamma]\!] = [\![AF\,\beta]\!] = \{3,4\}$

$= [\![\mu Z.\ \beta \vee [next]\, Z]\!]$

$[\![Z_0]\!] = \emptyset$

$[\![Z_1]\!] = [\![\beta \vee [next]\, Z_0]\!] =$

$= [\![\beta]\!] \vee PreA(next, [\![Z_0]\!])$

$\{4\} \quad \cup \quad \emptyset \quad = \{4\}$

$[\![Z_2]\!] = [\![\beta \vee [next]\, Z_1]\!]$

$[\![\beta]\!] \cup PreA(next, [\![Z_1]\!]) =$

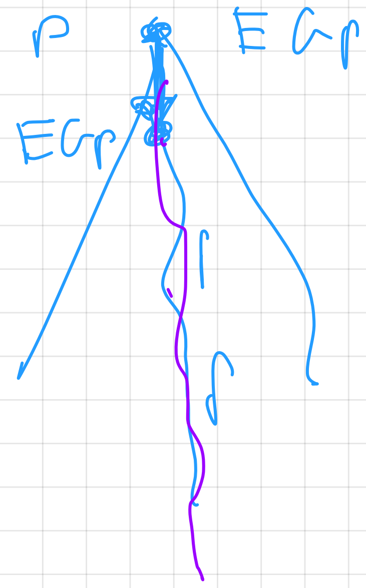
$\{4\} \cup \{3,4\} = \{3,4\}$

$$[\![ Z_3 ]\!] = [\![ B \vee [\mathit{next}]\, Z_2 ]\!]$$

$$= [\![ B ]\!] \cup \mathrm{Pre}_A(\mathit{next}, [\![ Z_2 ]\!])$$

$$= \{4\} \cup \{3,4\} = \{3,4\}$$

$$\boxed{EG\,p} \equiv p \wedge EX\,\boxed{EG\,p}$$



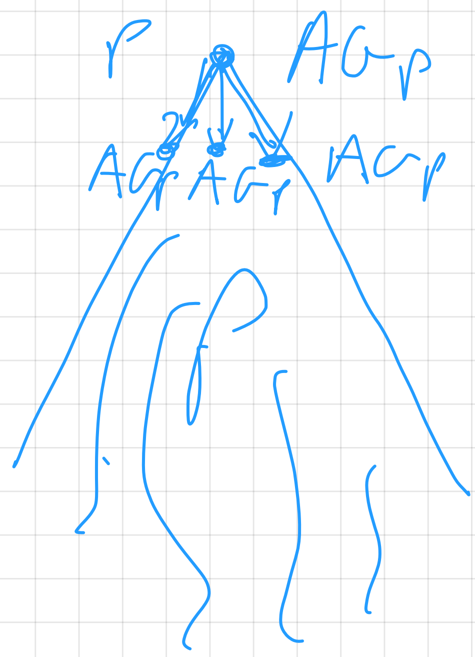

$$Z \equiv p \wedge EX\,Z$$

$$\text{gfp}\; Z \equiv p \wedge \langle next \rangle Z$$

$$\nu Z.\; p \wedge \langle next \rangle Z$$

$$t(EG\,\varphi) \rightsquigarrow \nu Z.\; t(\varphi) \wedge \langle next \rangle Z$$

$$\boxed{AG\,p} \equiv p \wedge AX\,\boxed{AG\,p}$$




$$Z \equiv p \wedge AX\,Z$$

$$\text{gfp}\; Z \equiv p \wedge [next]\, Z$$

$$\nu Z.\; p \wedge [next]\, Z$$

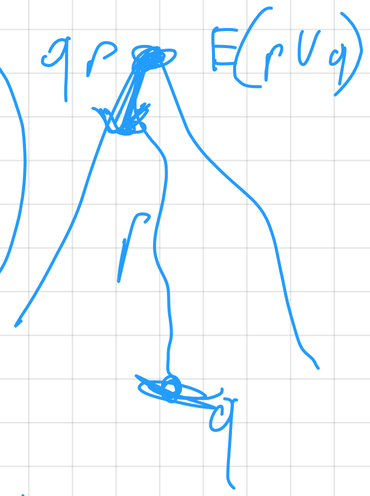
$$t(AG\,\varphi) \rightsquigarrow \nu Z.\; t(\varphi) \wedge [next]\, Z$$

$$\boxed{E(p \, U \, q)} \equiv q \vee (p \wedge EX \, \boxed{E(p \, U \, q)})$$

$$Z \equiv q \vee (p \wedge EX \, Z)$$

lfp $Z \equiv q \vee (p \wedge \langle next \rangle Z)$

$$\mu Z. \; q \vee (p \wedge \langle next \rangle Z)$$



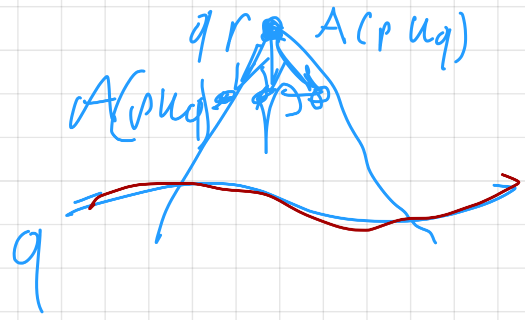
$$\vdash (E(\varphi_1 \, U \, \varphi_2)) \rightsquigarrow \mu Z. \; \vdash(\varphi_2) \vee \vdash(\varphi_1) \wedge \langle next \rangle Z$$

$$\boxed{A(p \, U \, q)} \equiv q \vee (p \wedge AX \, \boxed{A(p U q)})$$

$$Z \equiv q \vee (p \wedge AX \, Z)$$

lfp $Z \equiv q \vee (p \wedge [next] \, Z)$

$$\mu Z. \; q \vee (p \wedge [next] \, Z)$$



$$\vdash (A(\varphi_1 \, U \, \varphi_2)) \rightsquigarrow \mu Z. \; \vdash(\varphi_2) \vee \vdash(\varphi_1) \wedge [next] Z$$

# CTL Model Checking

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.*

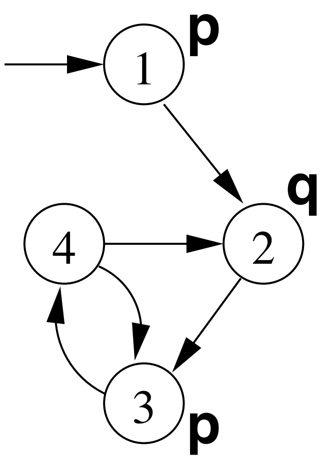
# Summary

- CTL Model Checking: General Ideas.
- CTL Model Checking: The Labeling Algorithm.
- Labeling Algorithm in Details.
- CTL Model Checking: Theoretical Issues.

# CTL Model Checking

CTL Model Checking is a formal verification technique s.t.

- The system is represented as a Kripke Model $\mathcal{KM}$:




- The property is expressed as a CTL formula $\varphi$, e.g.:

$$\mathbf{AG}(p \Rightarrow \mathbf{AF}q)$$

- The algorithm checks whether **all** the initial states, $s_0$, of the Kripke model satisfy the formula ($\mathcal{KM}, s_0 \models \varphi$).

# CTL M.C. Algorithm: General Ideas

The algorithm proceeds along two macro-steps:

1. Construct the set of states where the formula holds:
   $[[\varphi]] := \{s \in S : \mathcal{KM}, s \models \varphi\}$
   ($[[\varphi]]$ is called the denotation of $\varphi$);

2. Then compare the denotation with the set of initial states:
   $I \subseteq [[\varphi]]$ ?

# CTL M.C. Algorithm: General Ideas

To compute $[\![\varphi]\!]$ proceed "bottom-up" on the structure of the formula, computing $[\![\varphi_i]\!]$ for each subformula $\varphi_i$ of $\varphi$.

For example, to compute $[\![\mathbf{AG}(p \Rightarrow \mathbf{AF}q)]\!]$ we need to compute:

- $[\![q]\!]$,
- $[\![\mathbf{AF}q]\!]$,
- $[\![p]\!]$,
- $[\![p \Rightarrow \mathbf{AF}q]\!]$,
- $[\![\mathbf{AG}(p \Rightarrow \mathbf{AF}q)]\!]$

# CTL M.C. Algorithm: General Ideas

To compute each $[\![\varphi_i]\!]$ for generic subformulas:

- Handle boolean operators by standard set operations;
- Handle temporal operators $\mathbf{AX}$, $\mathbf{EX}$ by computing pre-images;
- Handle temporal operators $\mathbf{AG}$, $\mathbf{EG}$, $\mathbf{AF}$, $\mathbf{EF}$, $\mathbf{AU}$, $\mathbf{EU}$, by applying fixpoint operators.

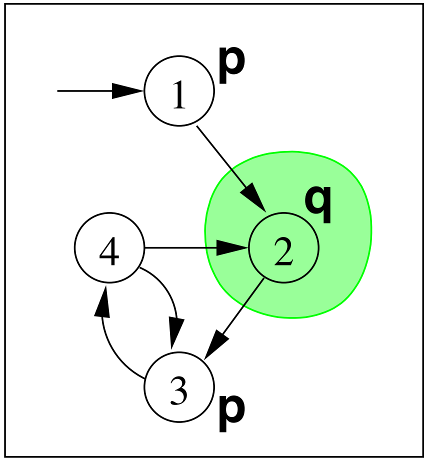
# Summary

- CTL Model Checking: General Ideas.
- CTL Model Checking: The Labeling Algorithm.
- Labeling Algorithm in Details.
- CTL Model Checking: Theoretical Issues.

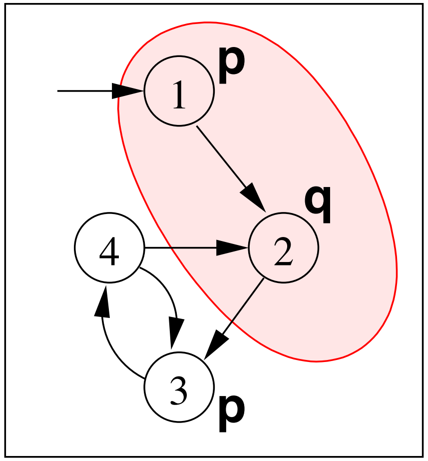
# The Labeling Algorithm: General Idea

- The Labeling Algorithm given a Kripke Model and a CTL formula outputs the set of states satisfying the formula.
- Main Idea: Label the states of the Kripke Model with the subformulas of $\varphi$ satisfied there.

# The Labeling Algorithm: An Example



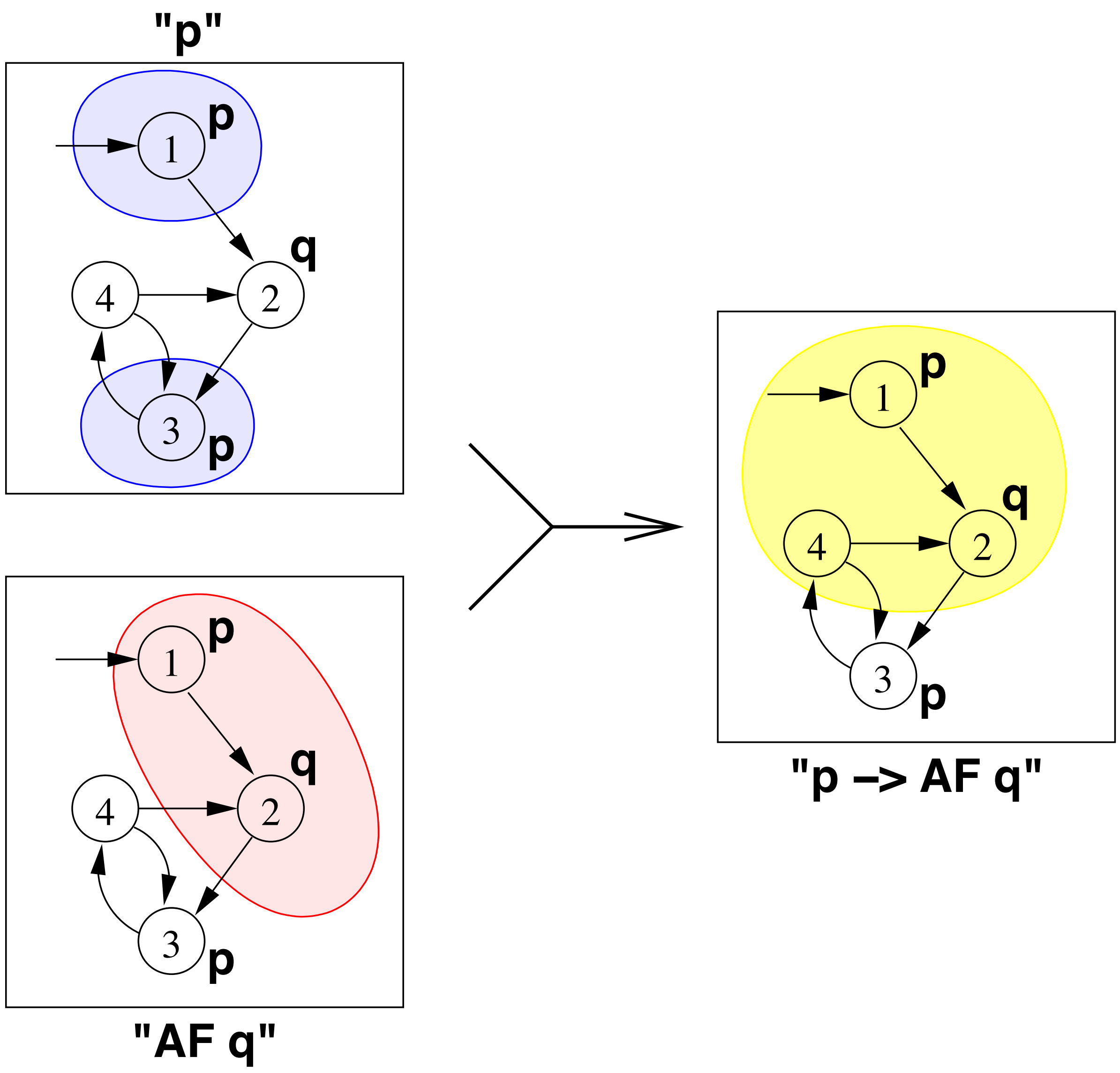

"q"



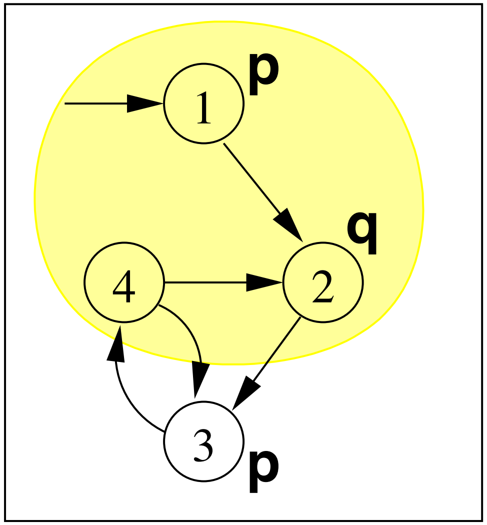

"AF q"

- $\mathbf{AF}q \equiv (q \vee \mathbf{AX}(\mathbf{AF}q))$
- $[\![\mathbf{AF}q]\!]$ can be computed as the union of:
  - $[\![q]\!] = \{2\}$
  - $[\![q \vee \mathbf{AX}q]\!] = \{2\} \cup \{1\} = \{1,2\}$
  - $[\![q \vee \mathbf{AX}(q \vee \mathbf{AX}q)]\!] = \{2\} \cup \{1\} = \{1,2\}$ (fixpoint).

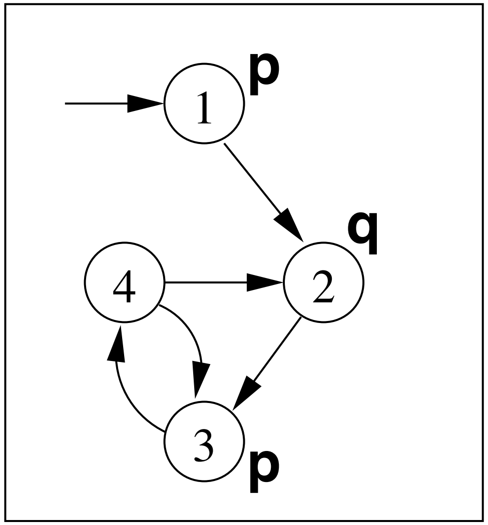
# The Labeling Algorithm: An Example



# The Labeling Algorithm: An Example




"p –> AF q"




"AG(p –> AF q)"

- $\mathbf{AG}\varphi \equiv (\varphi \wedge \mathbf{AX}(\mathbf{AG}\varphi))$
- $[\![\mathbf{AG}\varphi]\!]$ can be computed as the intersection of:
  - $[\![\varphi]\!] = \{1,2,4\}$
  - $[\![\varphi \wedge \mathbf{AX}\varphi]\!] = \{1,2,4\} \cap \{1,3\} = \{1\}$
  - $[\![\varphi \wedge \mathbf{AX}(\varphi \wedge \mathbf{AX}\varphi)]\!] = \{1,2,4\} \cap \{\} = \{\}$ (fixpoint)

# The Labeling Algorithm: An Example

- The set of states where the formula holds is empty, thus:
  - The initial state does not satisfy the property;
  - $\mathcal{KM} \not\models \mathbf{AG}(p \Rightarrow \mathbf{AF}q)$.

- Counterexample: A lazo-shaped path: $1, 2, \{3, 4\}^{\omega}$ (satisfying $\mathbf{EF}(p \wedge \mathbf{EG}\neg q)$)

# Summary

- CTL Model Checking: General Ideas.
- CTL Model Checking: The Labeling Algorithm.
- Labeling Algorithm in Details.
- CTL Model Checking: Theoretical Issues.

# The Labeling Algorithm: General Schema

- Assume $\varphi$ written in terms of $\neg$, $\wedge$, $\mathbf{EX}$, $\mathbf{EU}$, $\mathbf{EG}$ – minimal set of CTL operators
- The Labeling algorithm takes a CTL formula and a Kripke Model as input and returns the set of states satisfying the formula (i.e., the *denotation* of $\varphi$):
  1. For every $\varphi_i \in Sub(\varphi)$, find $[\![\varphi_i]\!]$;
  2. Compute $[\![\varphi]\!]$ starting from $[\![\varphi_i]\!]$;
  3. Check if $I \subseteq [\![\varphi]\!]$.
- Subformulas $Sub(\varphi)$ of $\varphi$ are checked bottom-up
- To compute each $[\![\varphi_i]\!]$: if the main operator of $\varphi_i$ is a
  - *Boolean Operator*: apply standard set operations;
  - *Temporal Operator*: apply recursive rules until a fixpoint is reached.

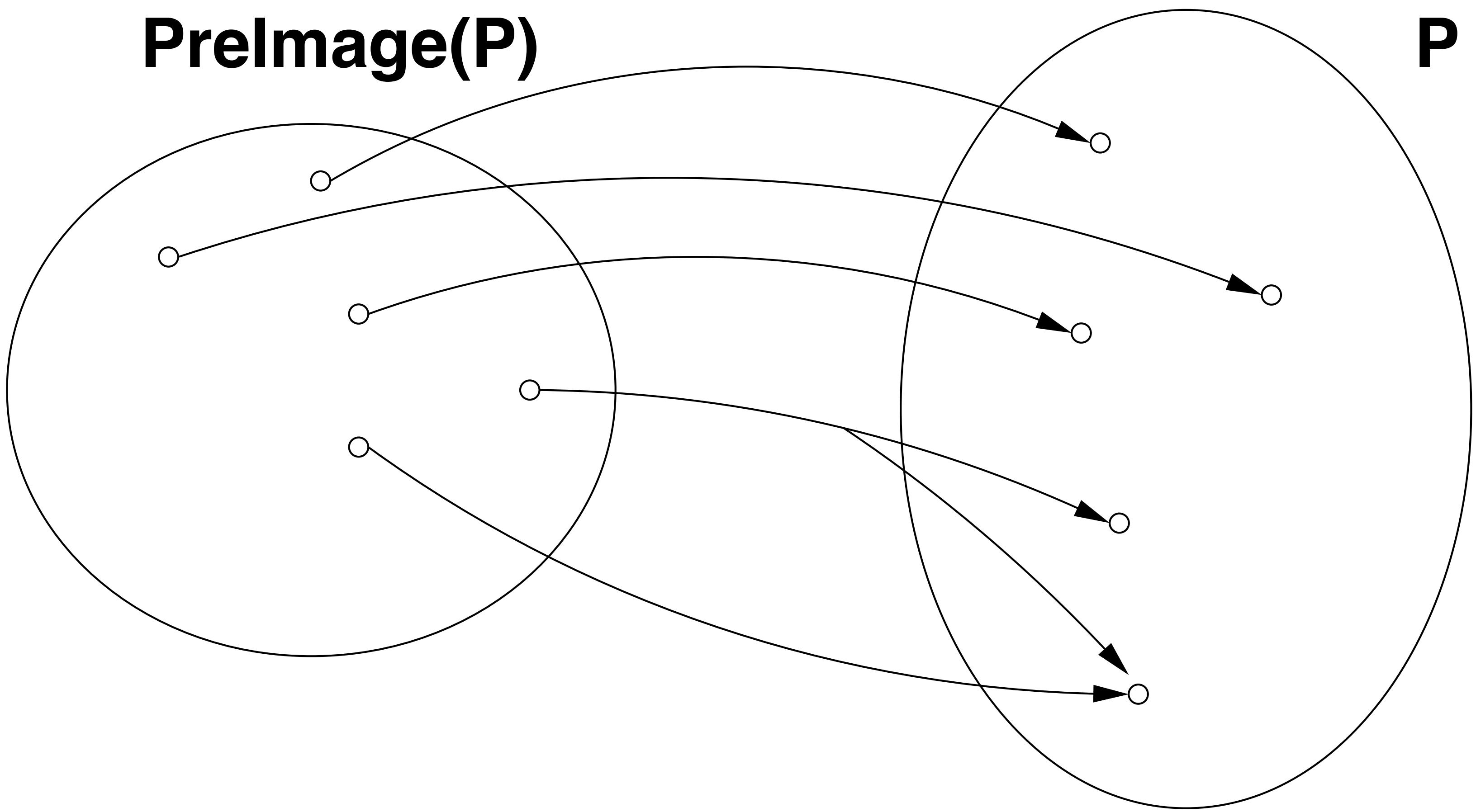
# Denotation of Formulas: The Boolean Case

Let $\mathcal{KM} = \langle S, I, R, L, \Sigma \rangle$ be a Kripke Model.

$$
\begin{aligned}
[\![false]\!] &= \{\} \\
[\![true]\!] &= S \\
[\![p]\!] &= \{s \mid p \in L(s)\} \\
[\![\neg\varphi_1]\!] &= S \setminus [\![\varphi_1]\!] \\
[\![\varphi_1 \wedge \varphi_2]\!] &= [\![\varphi_1]\!] \cap [\![\varphi_2]\!]
\end{aligned}
$$

# Denotation of Formulas: The EX Case

- $[\![\mathbf{EX}\varphi]\!] = \{s \in S \mid \exists s'.\langle s, s'\rangle \in R \text{ and } s' \in [\![\varphi]\!]\}$
- $[\![\mathbf{EX}\varphi]\!]$ is said to be the Pre-image of $[\![\varphi]\!]$ ($\textsc{Pre}([\![\varphi]\!])$).
- Key step of every CTL M.C. operation.



# Denotation of Formulas: The EG Case

- From the semantics of the $\Box$ temporal operator:

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

- Then, the following equivalence holds:

$$\mathbf{EG}\varphi \equiv \varphi \wedge \mathbf{EX}(\mathbf{EG}\varphi)$$

- To compute $[\![\mathbf{EG}\varphi]\!]$ we can apply the following recursive definition:

$$[\![\mathbf{EG}\varphi]\!] = [\![\varphi]\!] \cap \textsc{Pre}([\![\mathbf{EG}\varphi]\!])$$

# Denotation of Formulas: The EG Case

- We can compute $X := [\![\mathbf{EG}\varphi]\!]$ inductively as follows:

$$
\begin{aligned}
X_1 &:= [\![\varphi]\!] \\
X_2 &:= X_1 \cap \textsc{Pre}(X_1) \\
&\ldots \\
X_{j+1} &:= X_j \cap \textsc{Pre}(X_j)
\end{aligned}
$$

- When $X_n = X_{n+1}$ we reach a fixpoint and we stop.
- **Termination.** Since $X_{j+1} \subseteq X_j$ for every $j \geq 0$, thus a fixed point always exists (Knaster-Tarski's theorem).

# Denotation of Formulas: The EU Case

- From the semantics of the $\mathcal{U}$ temporal operator:

  $$\varphi\, \mathcal{U}\, \psi \equiv \psi \vee (\varphi \wedge \bigcirc(\varphi\, \mathcal{U}\, \psi))$$

- Then, the following equivalence holds:

  $$(\varphi \mathbf{EU} \psi) \equiv \psi \vee (\varphi \wedge \mathbf{EX}(\varphi \mathbf{EU} \psi))$$

- To compute $[\![(\varphi \mathbf{EU} \psi)]\!]$ we can apply the following recursive definition:

  $$[\![(\varphi \mathbf{EU} \psi)]\!] = [\![\psi]\!] \cup ([\![\varphi]\!] \cap \textsc{Pre}([\![(\varphi \mathbf{EU} \psi)]\!]))$$

# Denotation of Formulas: The EU Case

- We can compute $X := [\![(\varphi \mathbf{EU} \psi)]\!]$ inductively as follows:

$$\begin{array}{lll} X_1 & := & [\![\psi]\!] \\ X_2 & := & X_1 \cup ([\![\varphi]\!] \cap \textsc{Pre}(X_1)) \\ \ldots & & \\ X_{j+1} & := & X_j \cup ([\![\varphi]\!] \cap \textsc{Pre}(X_j)) \end{array}$$

- When $X_n = X_{n+1}$ we reach a fixpoint and we stop.
- **Termination.** Since $X_{j+1} \supseteq X_j$ for every $j \geq 0$, thus a fixed point always exists (Knaster-Tarski's theorem).

# The Pseudo-Code

We assume the Kripke Model to be a global variable:

FUNCTION Label($\varphi$) {
  **case** $\varphi$ **of**
    $true$: **return** $S$;
    $false$: **return** $\{\}$;
    an atom $p$: **return** $\{s \in S \mid p \in L(s)\}$;
    $\neg\varphi_1$: **return** $S \setminus$Label($\varphi_1$);
    $\varphi_1 \wedge \varphi_2$: **return** Label($\varphi_1$)$\cap$Label($\varphi_2$);
    $\mathbf{EX}\varphi_1$: **return** PRE(Label($\varphi_1$));
    $(\varphi_1 \mathbf{EU} \varphi_2)$: **return** Label_EU(Label($\varphi_1$),Label($\varphi_2$));
    $\mathbf{EG}\varphi_1$: **return** Label_EG(Label($\varphi_1$));
  **end case**
}

# PreImage

$[\![\mathbf{EX}\varphi]\!] = \textsc{Pre}([\![\varphi]\!]) = \{s \in S \mid \exists s'.\langle s, s'\rangle \in R \text{ and } s' \in [\![\varphi]\!]\}$

$\textsc{Function Pre}([\![\varphi]\!])\{$

**var** $X$;

$X := \{\}$;

**for each** $s' \in [\![\varphi]\!]$ **do**

**for each** $s \in S$ **such that** $\langle s, s'\rangle \in R$ **do**

$X := X \cup \{s\}$;

**return** $X$

$\}$

# Label_EG

$[\![\mathbf{EG}\varphi]\!] = [\![\varphi]\!] \cap \textsc{Pre}([\![\mathbf{EG}\varphi]\!])$

$\textsc{Function Label\_EG}([\![\varphi]\!])\{$

$\quad$ **var** $X, OLD\text{-}X$;

$\quad X := [\![\varphi]\!]$;

$\quad OLD\text{-}X := \emptyset$;

$\quad$ **while** $X \neq OLD\text{-}X$

$\quad$ **begin**

$\qquad OLD\text{-}X := X$;

$\qquad X := X \cap \textsc{Pre}(X)$

$\quad$ **end**

$\quad$ **return** $X$

$\}$

# Label_EU

$$[\![(\varphi \mathbf{EU} \psi)]\!] = [\![\psi]\!] \cup ([\![\varphi]\!] \cap \textsc{Pre}([\![(\varphi \mathbf{EU} \psi)]\!]))$$

FUNCTION LABEL_EU($[\![\varphi]\!]$,$[\![\psi]\!]$){

**var** $X, \mathit{OLD}\text{-}X$;

$X := [\![\psi]\!]$;

$\mathit{OLD}\text{-}X := S$;

**while** $X \neq \mathit{OLD}\text{-}X$

**begin**

$\mathit{OLD}\text{-}X := X$;

$X := X \cup ([\![\varphi]\!] \cap \textsc{Pre}(X))$

**end**

**return** $X$

}

# Summary

- CTL Model Checking: General Ideas.
- CTL Model Checking: The Labeling Algorithm.
- Labeling Algorithm in Details.
- CTL Model Checking: Theoretical Issues.

# Correctness and Termination

- The Labeling algorithm works recursively on the structure $\varphi$.
- For most of the logical constructors the algorithm does the correct things according to the semantics of CTL.
- To prove that the algorithm is *Correct* and *Terminating* we need to prove the correctness and termination of both **EG** and **EU** operators.

# Monotone Functions and Fixpoints

**Definition.** Let $S$ be a set and $F$ a function, $F : 2^S \rightarrow 2^S$, then:

1. $F$ is monotone iff $X \subseteq Y$ then $F(X) \subseteq F(Y)$;
2. A subset $X$ of $S$ is called a fixpoint of $F$ iff $F(X) = X$;
3. $X$ is a least fixpoint (LFP) of $F$, written $\mu X.F(X)$, iff, for every other fixpoint $Y$ of $F$, $X \subseteq Y$
4. $X$ is a greatest fixpoint (GFP) of $F$, written $\nu X.F(X)$, iff, for every other fixpoint $Y$ of $F$, $Y \subseteq X$

**Example.** Let $S = \{s_0, s_1\}$ and $F(X) = X \cup \{s_0\}$.

# Knaster-Tarski Theorem

**Notation:** $F^i(X)$ means applying $F$ $i$-times, i.e., $F(F(\ldots F(X)\ldots))$.

**Theorem[Knaster-Tarski].** Let $S$ be a finite set with $n+1$ elements. If $F : 2^S \to 2^S$ is a monotone function then:

1. $\mu X.F(X) \equiv F^{n+1}(\emptyset)$;
2. $\nu X.F(X) \equiv F^{n+1}(S)$.

# Correctness and Termination: EG Case

The function $\textsc{Label\_EG}$ computes:

$$[\![\mathbf{EG}\varphi]\!] = [\![\varphi]\!] \cap \textsc{Pre}([\![\mathbf{EG}\varphi]\!])$$

applying the semantic equivalence:

$$\mathbf{EG}\varphi \equiv \varphi \wedge \mathbf{EX}(\mathbf{EG}\varphi)$$

Thus, $[\![\mathbf{EG}\varphi]\!]$ is the fixpoint of the function:

$$F(X) = [\![\varphi]\!] \cap \textsc{Pre}(X)$$

# Correctness and Termination: EG Case

**Theorem.** Let $F(X) = [\![\varphi]\!] \cap \textsc{Pre}(X)$, and let $S$ have $n+1$ elements. Then:

1. $F$ is monotone;
2. $[\![\mathbf{EG}\varphi]\!]$ is the greatest fixpoint of $F$.

# Correctness and Terminationpr: EU Case

The function $\textsc{Label\_EU}$ computes:

$$[\![(\varphi \mathbf{EU} \psi)]\!] = [\![\psi]\!] \cup ([\![\varphi]\!] \cap \textsc{Pre}([\![(\varphi \mathbf{EU} \psi)]\!]))$$

applying the semantic equivalence:

$$(\varphi \mathbf{EU} \psi) \equiv \psi \vee (\varphi \wedge \mathbf{EX}(\varphi \mathbf{EU} \psi))$$

Thus, $[\![(\varphi \mathbf{EU} \psi)]\!]$ is the fixpoint of the function:

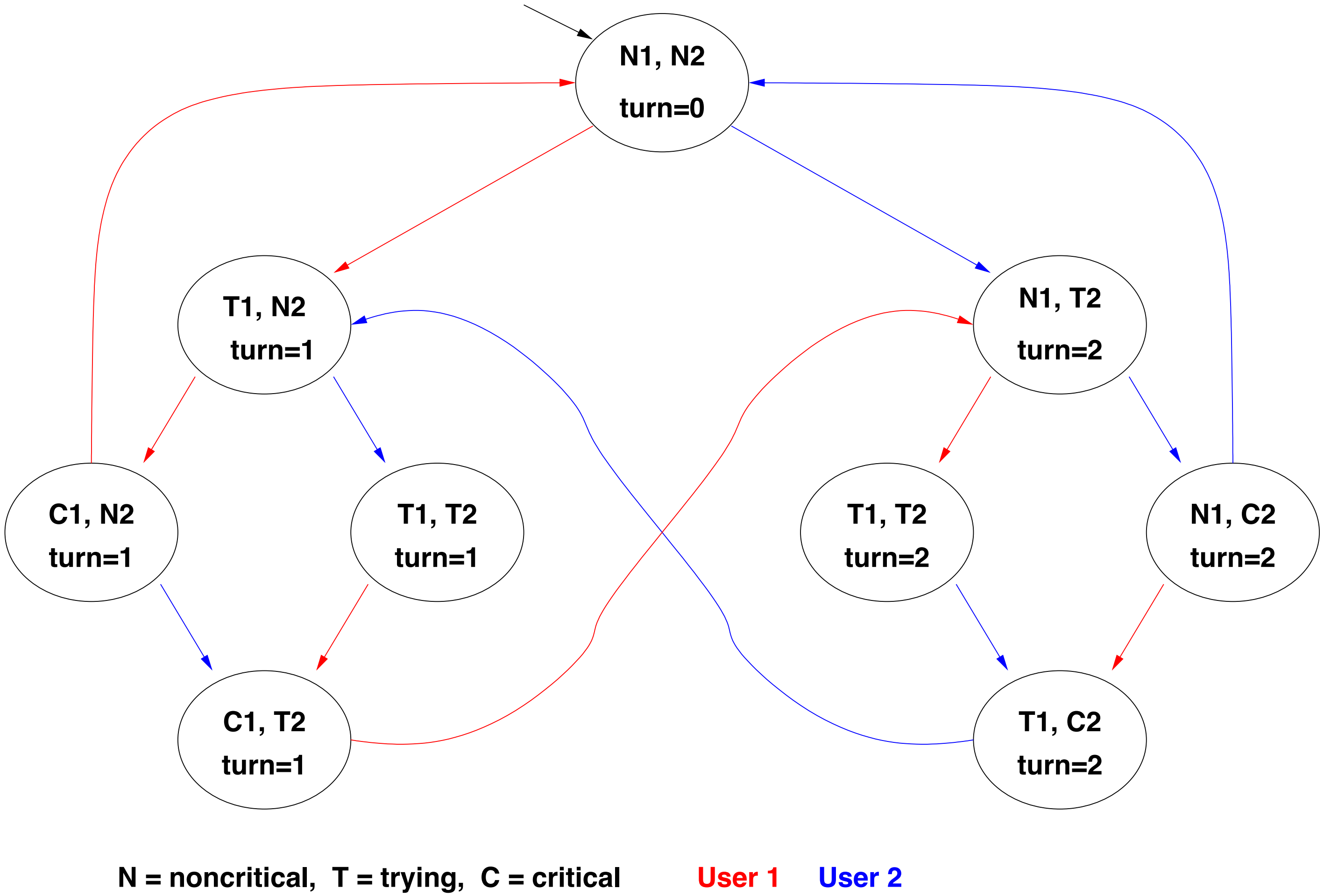
$$F(X) = [\![\psi]\!] \cup ([\![\varphi]\!] \cap \textsc{Pre}(X))$$

# Correctness and Termination: EU Case

**Theorem.** Let $F(X) = [\![\psi]\!] \cup ([\![\varphi]\!] \cap \textsc{Pre}(X))$, and let $S$ have $n+1$ elements. Then:

1. $F$ is monotone;
2. $[\![(\varphi \mathbf{EU} \psi)]\!]$ is the least fixpoint of $F$.

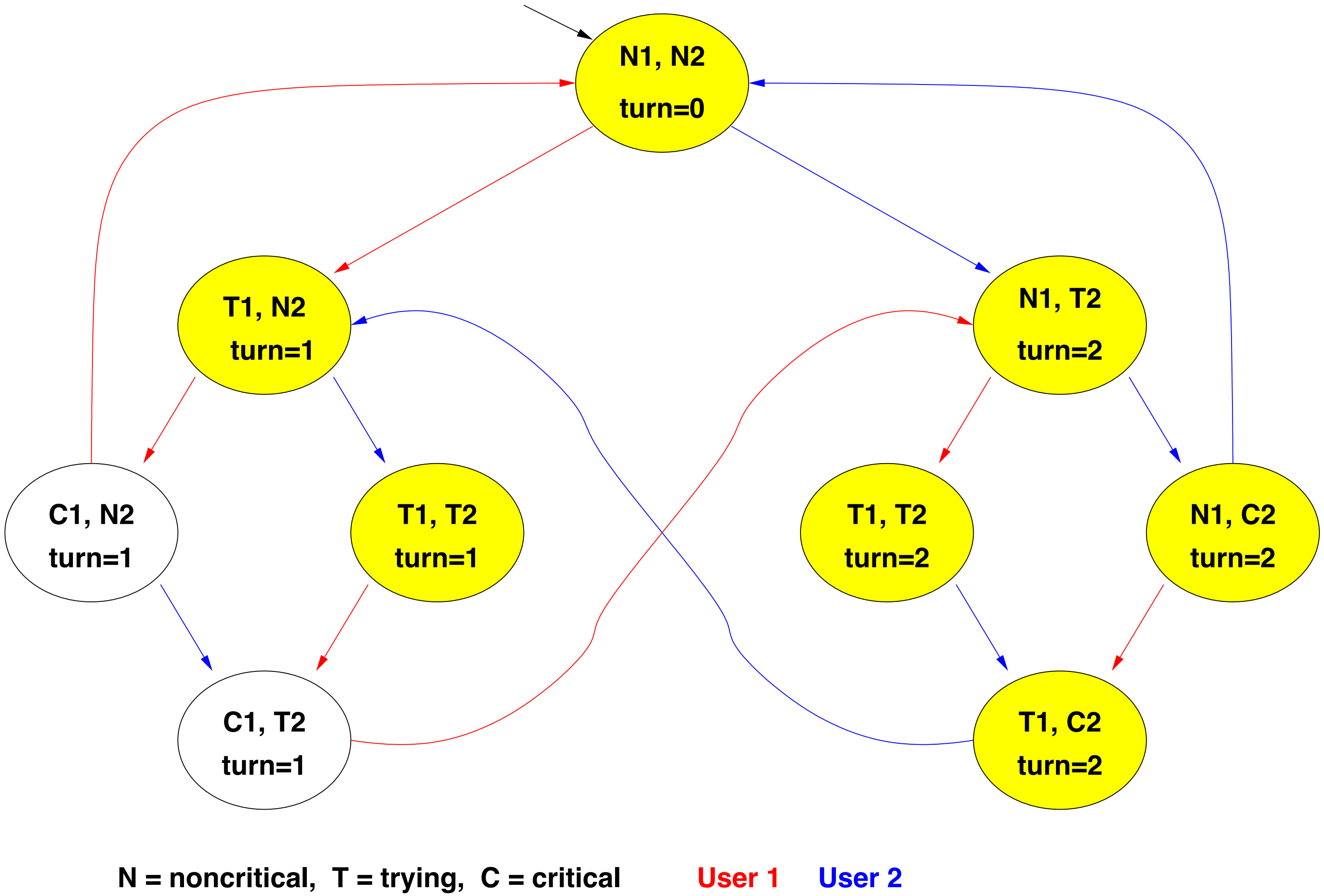
# Example 1: fairness




$$M \models \mathbf{AGAF}C_1 \; ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \; ?$$

# Example 1: fairness

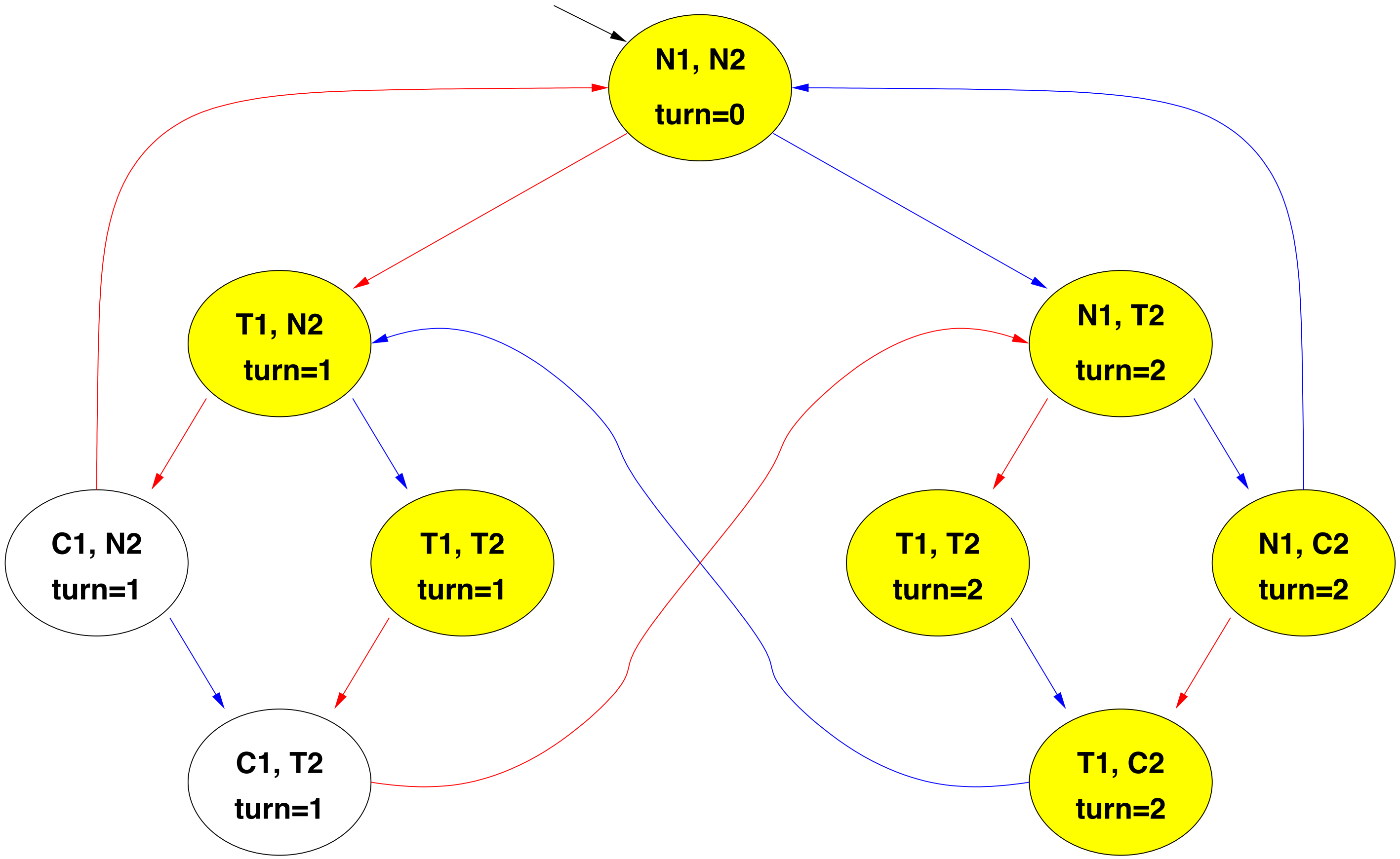
$[\neg C_1]$




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

$[\mathbf{EG}\neg C_1]$, step 0:

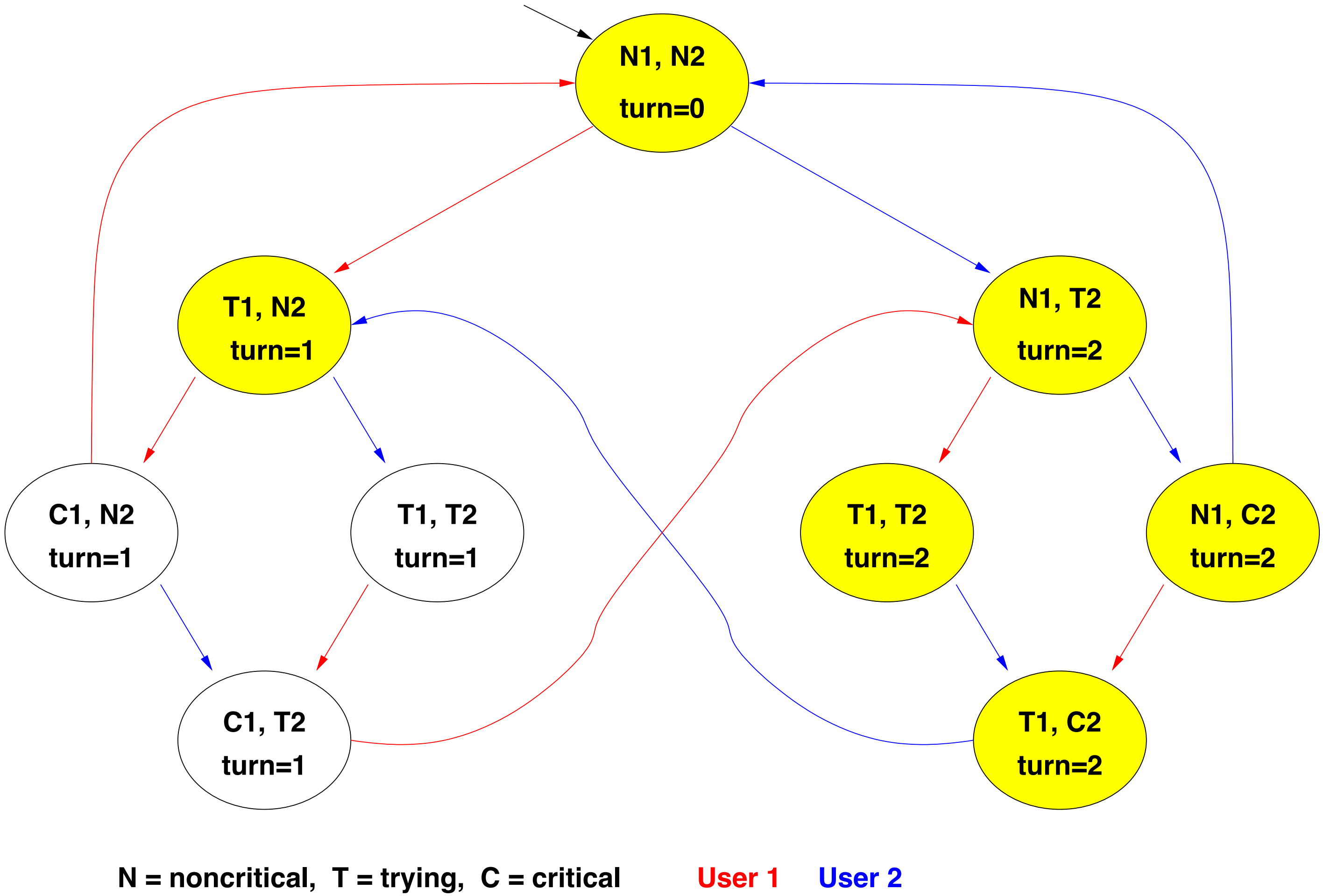



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AGAF}C_1 \;? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \;?$$

# Example 1: fairness

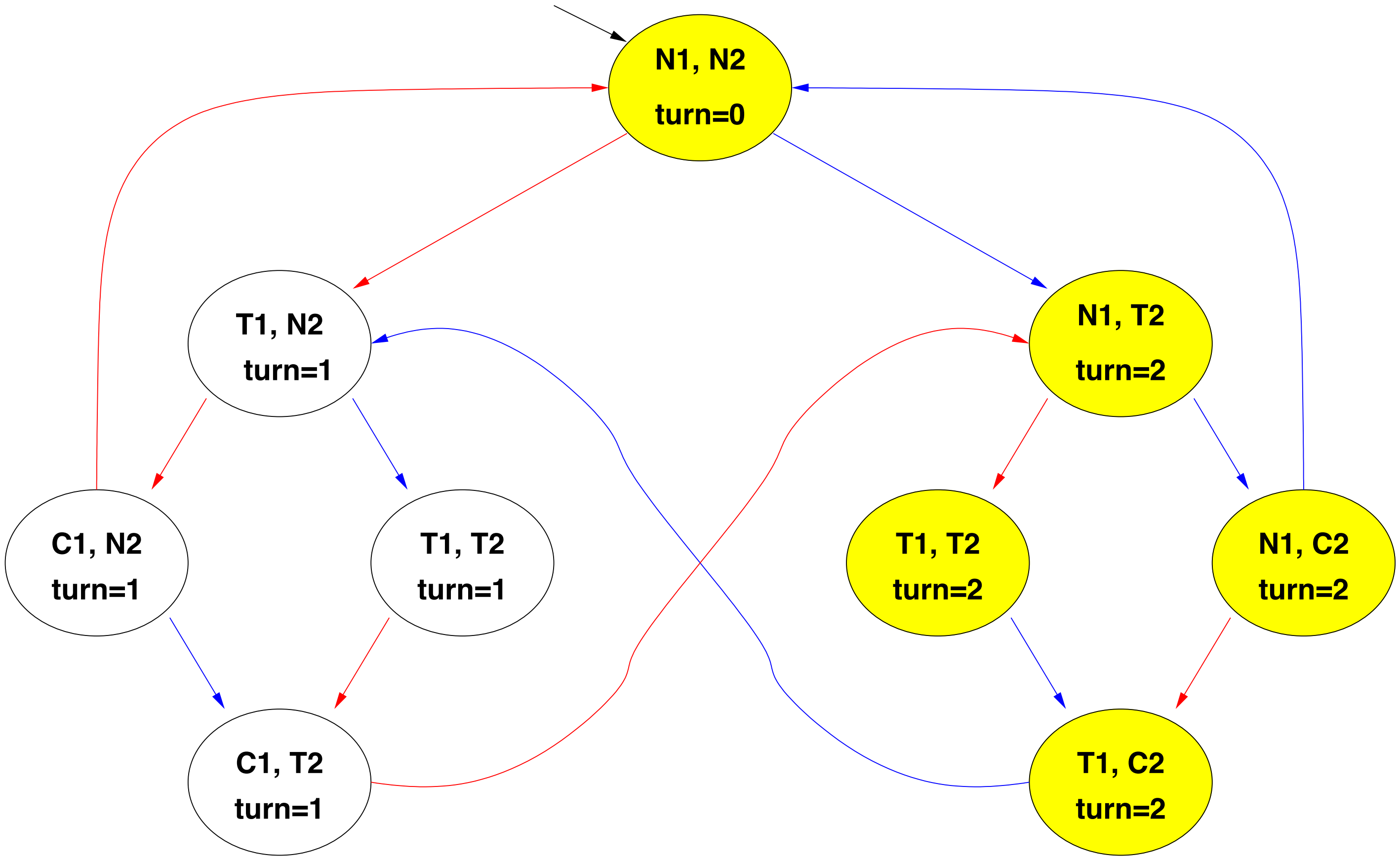
$[\mathbf{EG}\neg C_1]$, step 1:




$M \models \mathbf{AGAF}C_1$ ? $\Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1$ ?

# Example 1: fairness

$[\mathbf{EG}\neg C_1]$, step 2:

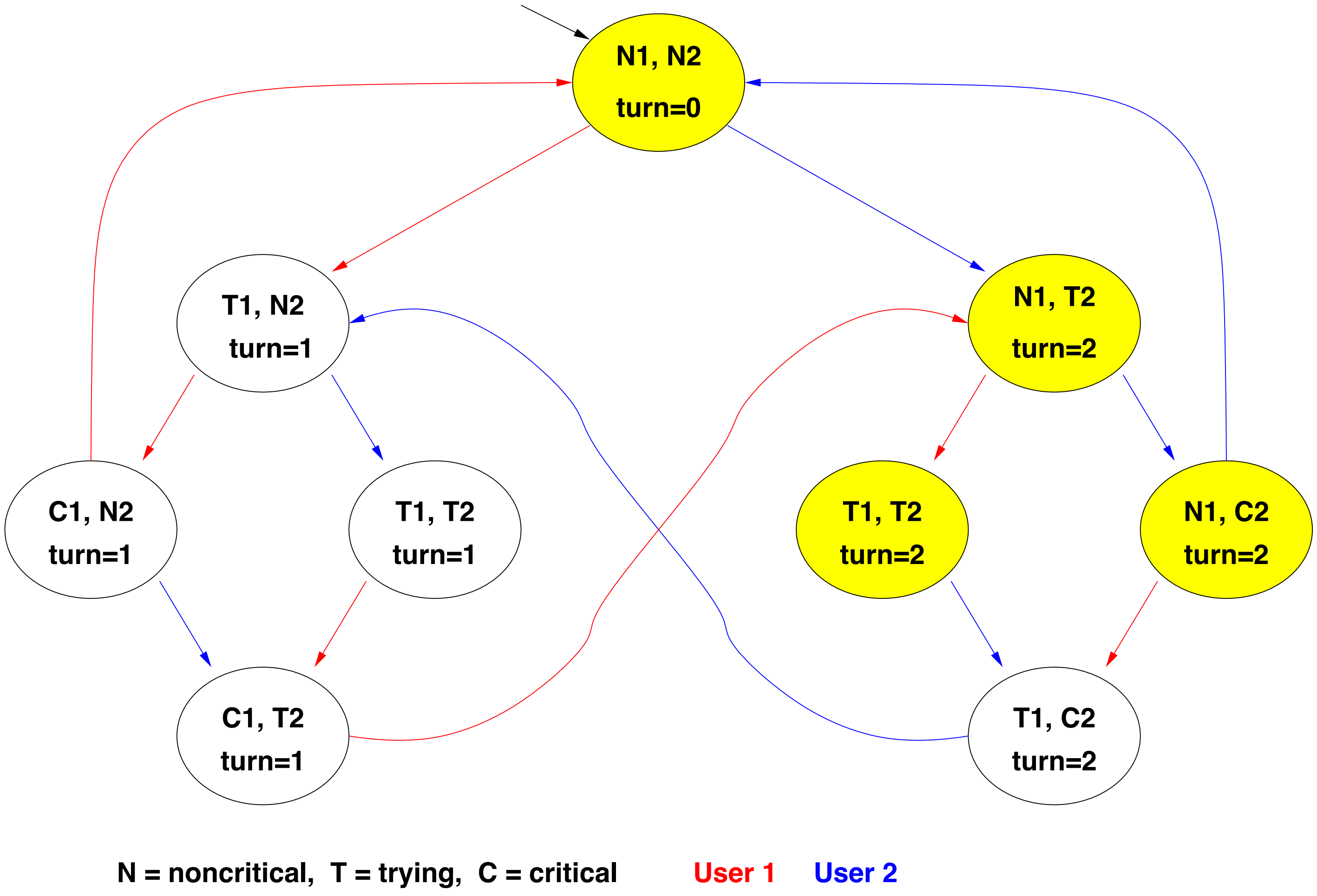



N = noncritical, T = trying, C = critical User 1 User 2

$M \models \mathbf{AGAF}C_1\ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1\ ?$

# Example 1: fairness

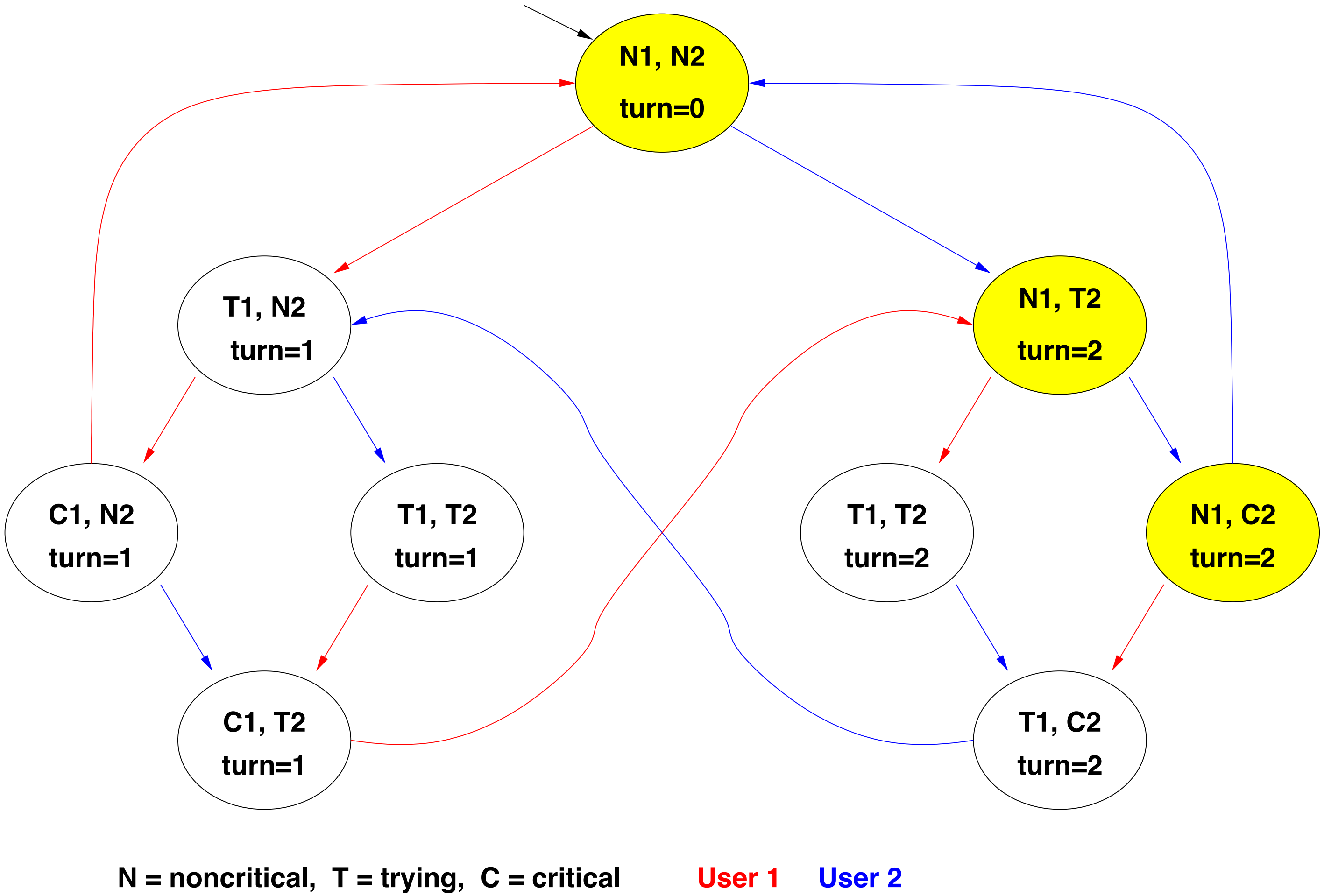
$[\mathbf{EG}\neg C_1]$, step 3:




$M \models \mathbf{AGAF}C_1 \; ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \; ?$

# Example 1: fairness

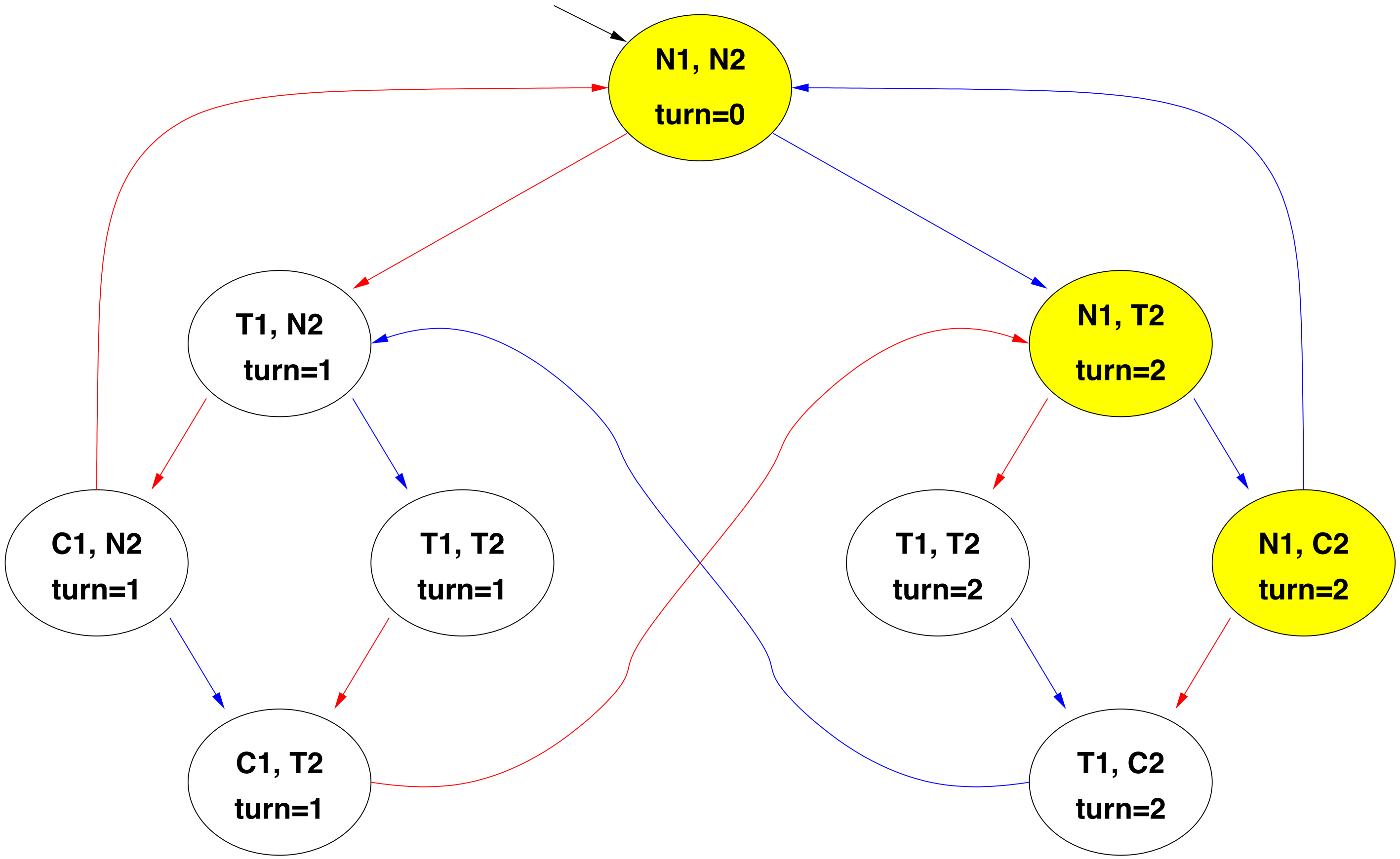
$[\mathbf{EG}\neg C_1]$, step 4:




$$M \models \mathbf{AGAF}C_1\ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1\ ?$$

# Example 1: fairness

$[\mathbf{EG}\neg C_1]$, FIXPOINT!

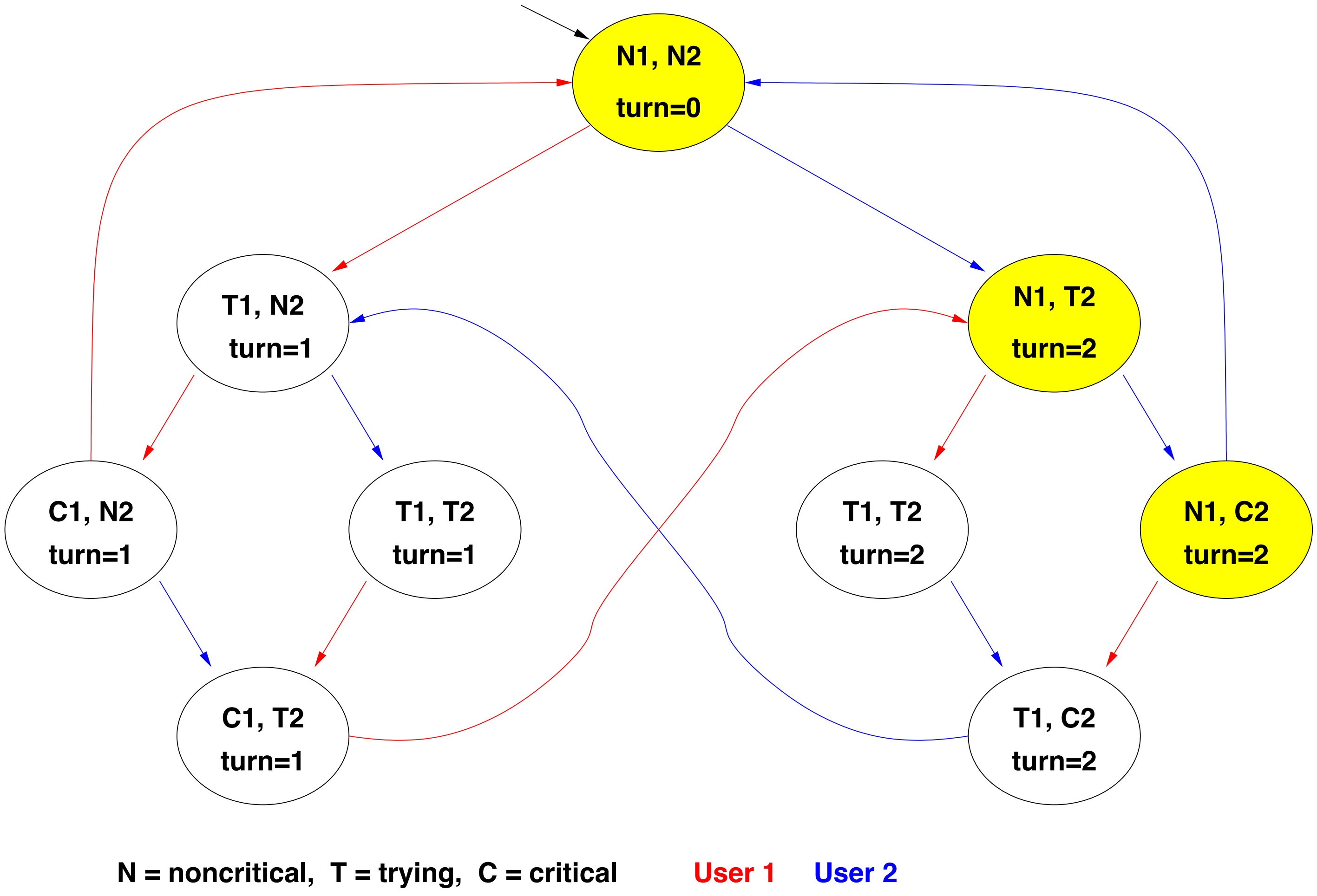



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AGAF}C_1 \text{ ?} \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \text{ ?}$$

# Example 1: fairness

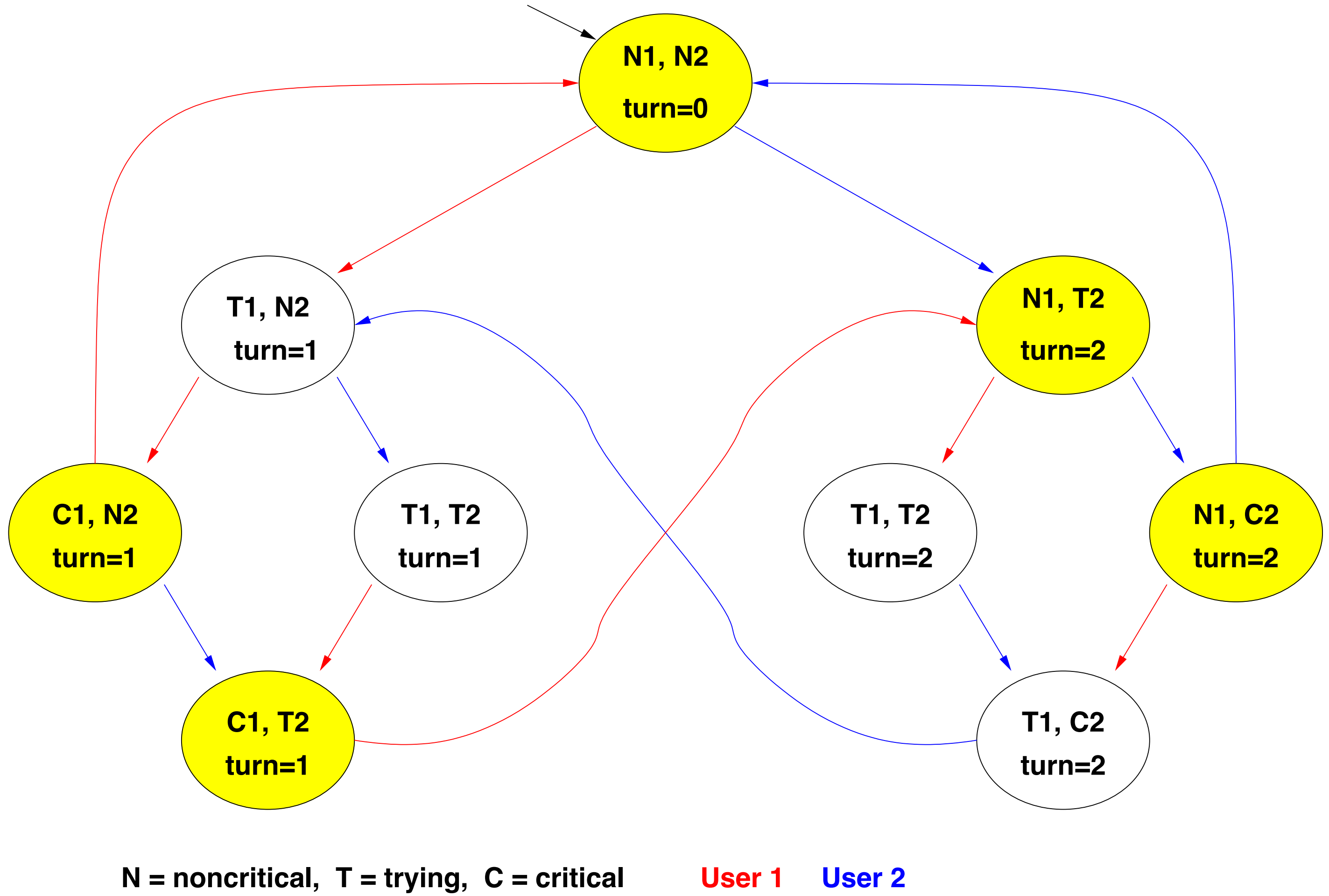
## $[\mathbf{EFEG}\neg C_1]$, STEP 0




$$M \models \mathbf{AGAF}C_1 \; ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \; ?$$

# Example 1: fairness

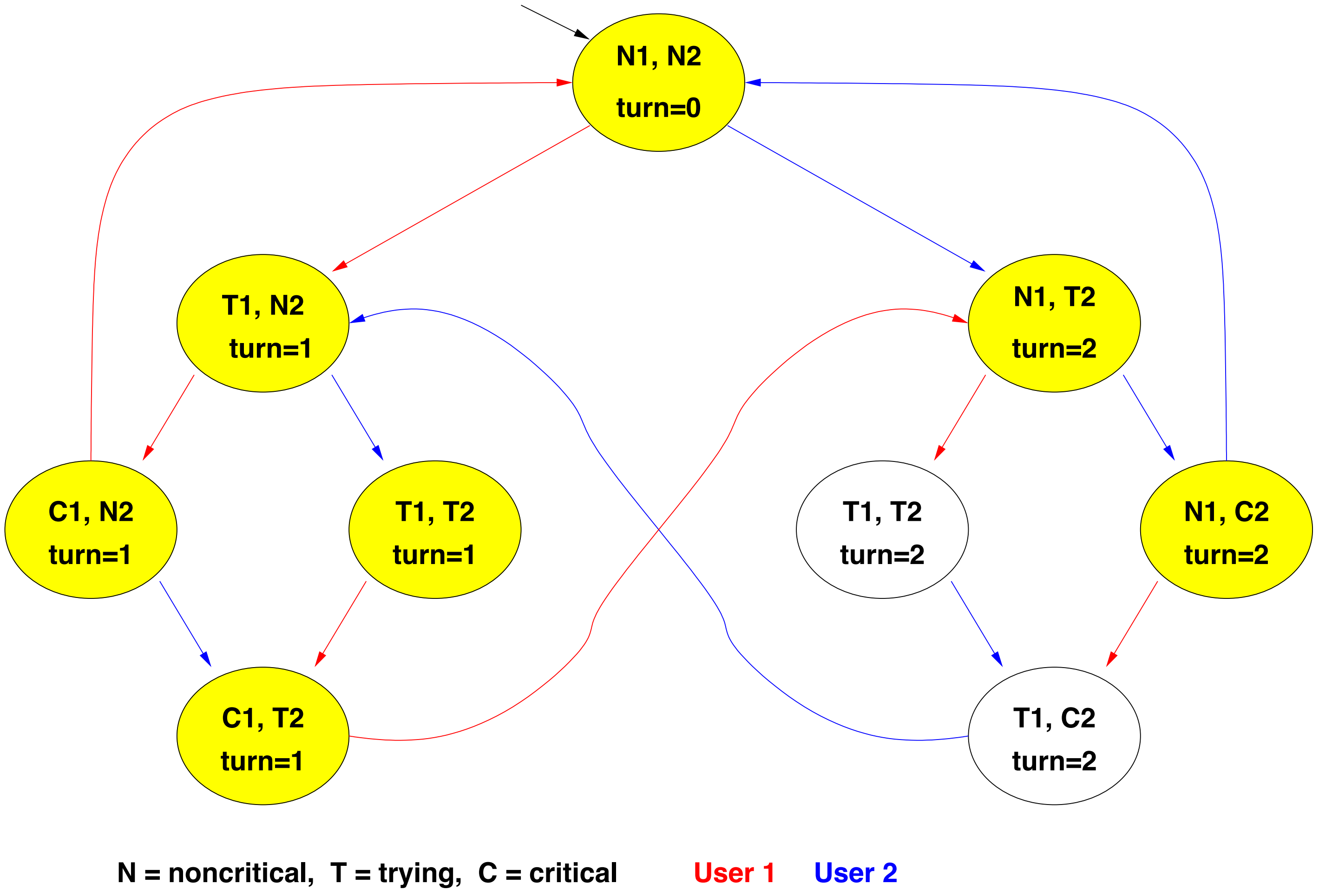
## $[\mathbf{EFEG}\neg C_1]$, STEP 1




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

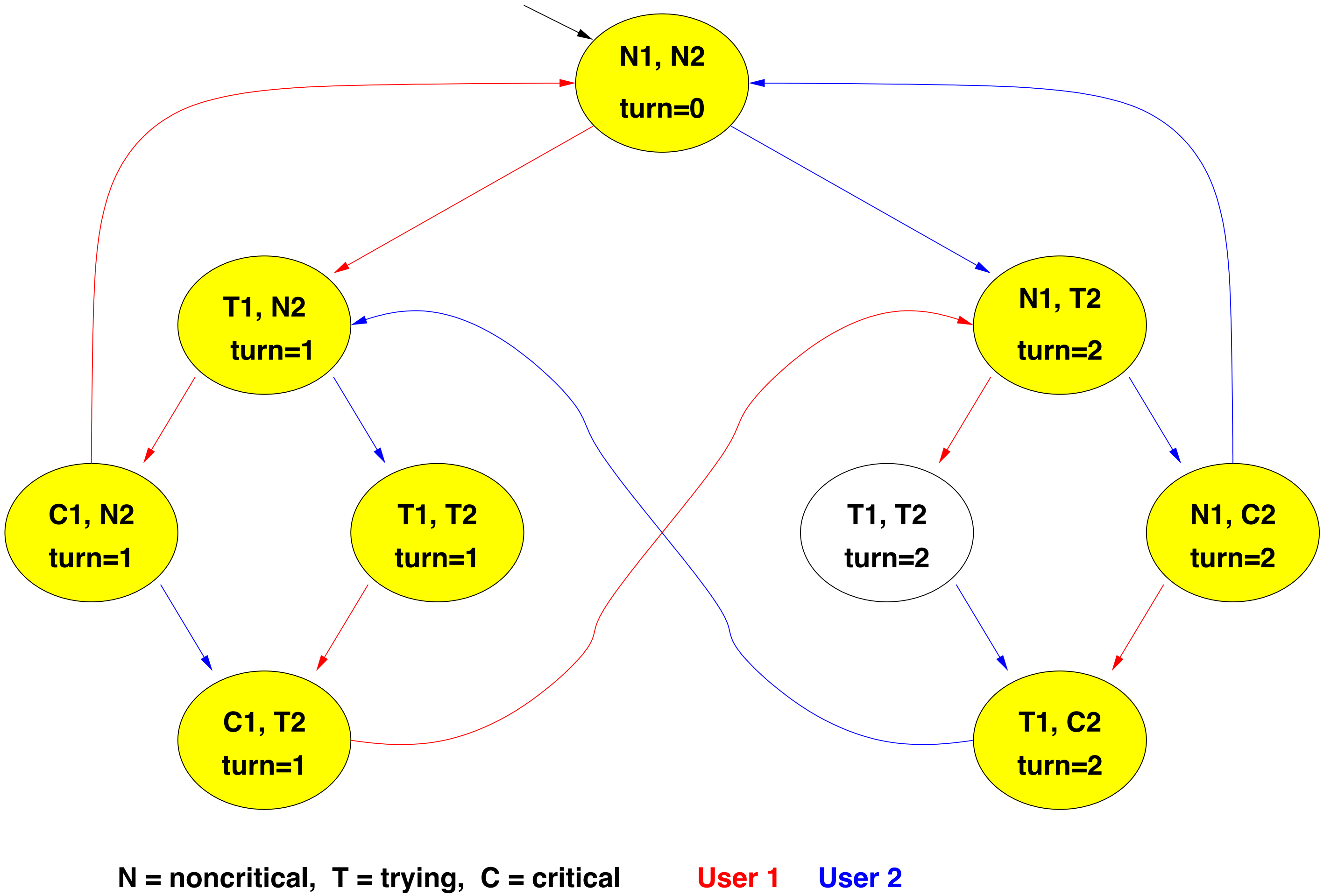
## $[\mathbf{EFEG}\neg C_1]$, STEP 2




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

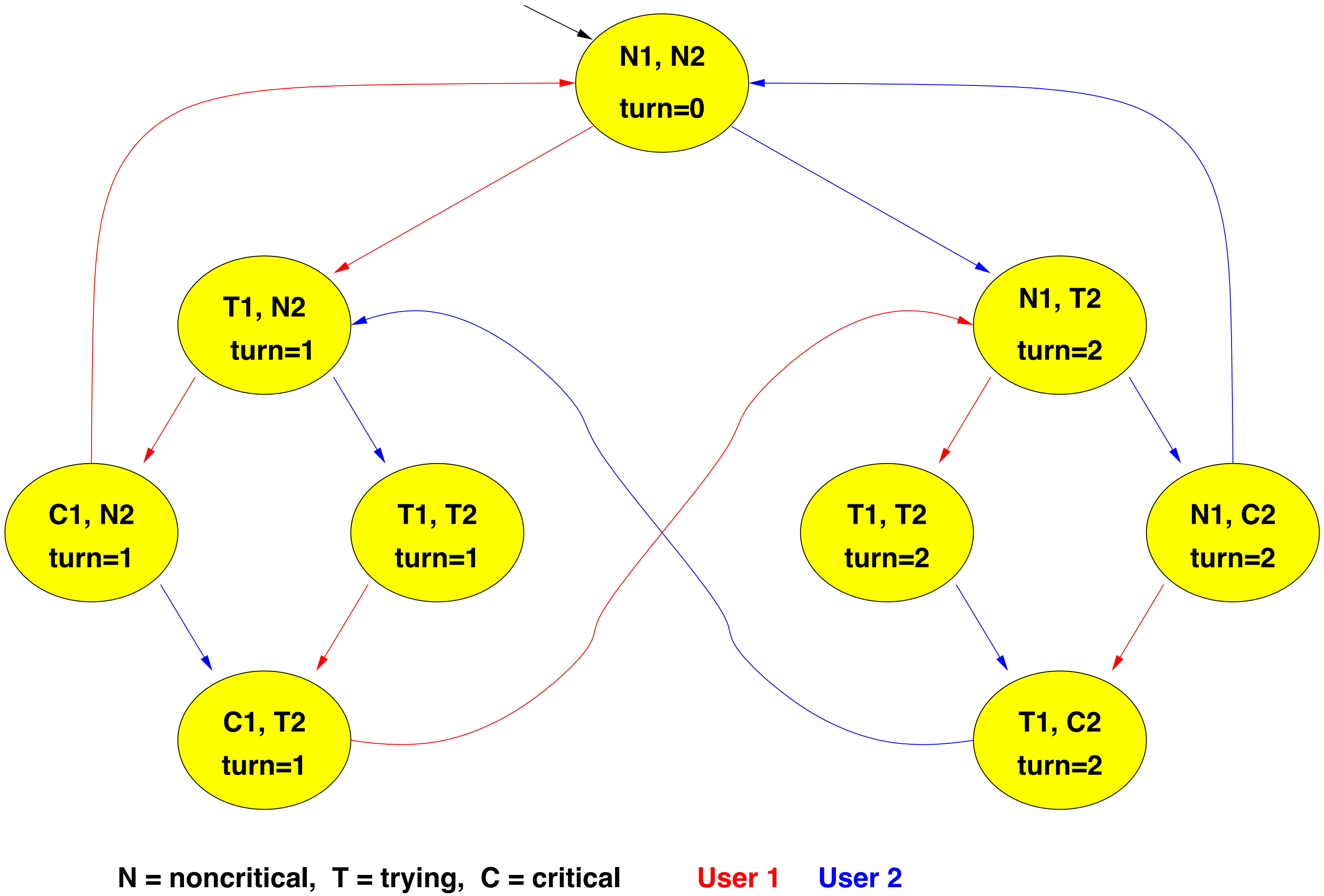
## $[\mathbf{EFEG}\neg C_1]$, STEP 3




$$M \models \mathbf{AGAF}C_1\ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1\ ?$$

# Example 1: fairness

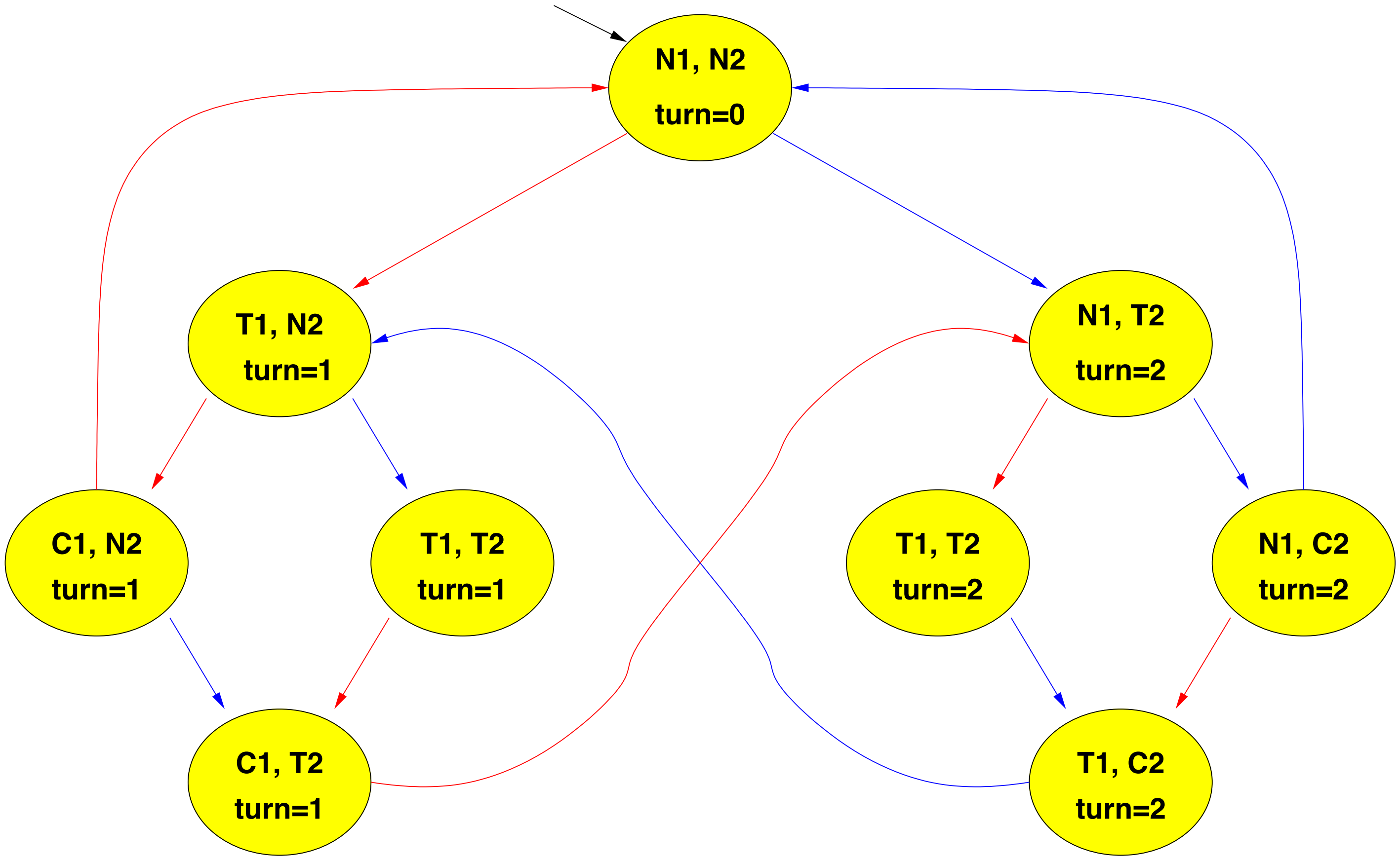
## $[\mathbf{EFEG}\neg C_1]$, STEP 4




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

$[\mathbf{EFG}\neg C_1]$, FIXPOINT!

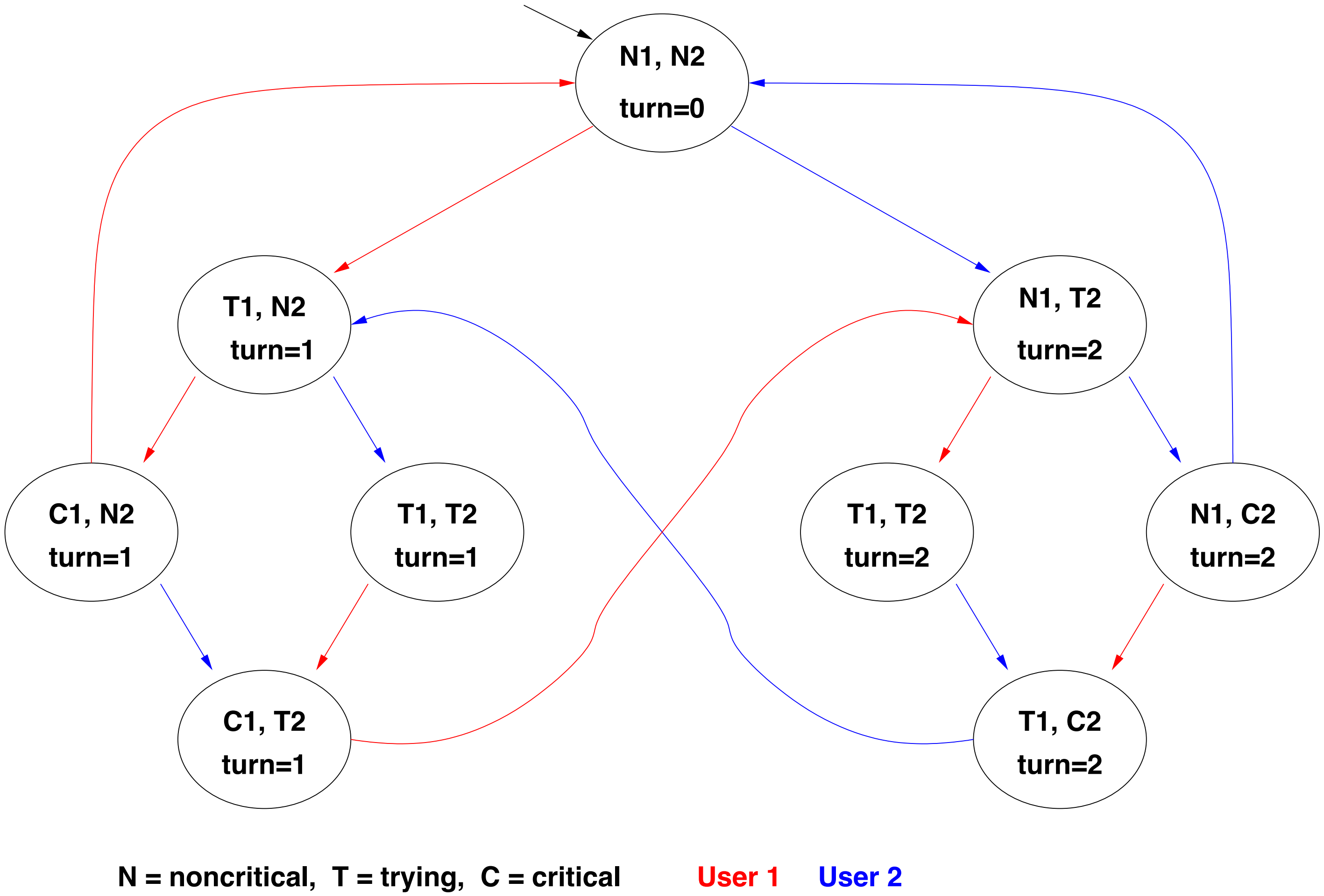



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AGAF}C_1\ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1\ ?$$

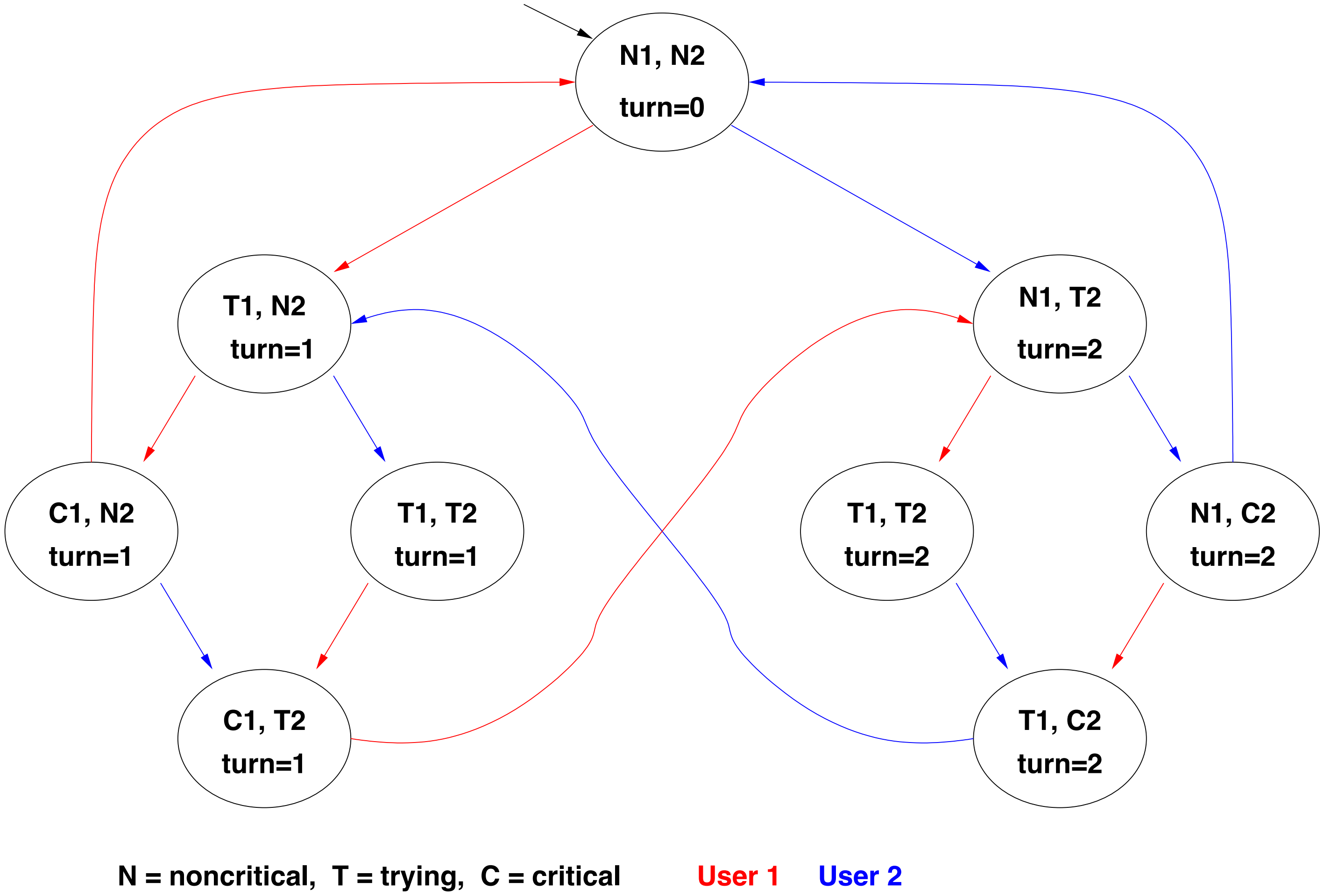
# Example 1: fairness

$[\neg \mathbf{EFEG}\neg C_1]$




$M \models \mathbf{AGAF}C_1$ ? $\Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1$ ? $\Longrightarrow$ NO!

# Example 2: liveness

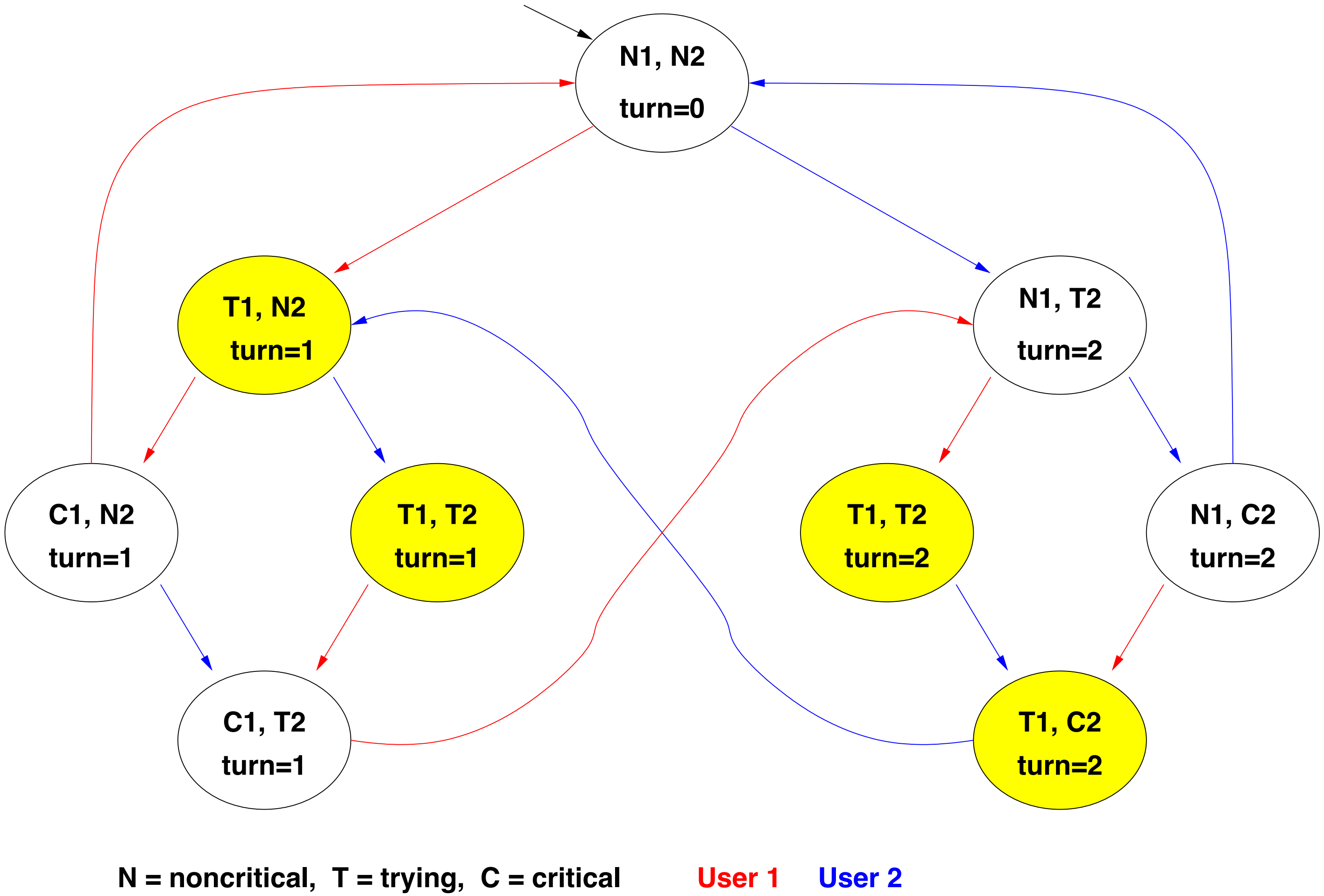



$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)\ ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)\ ?$$

# Example 2: liveness

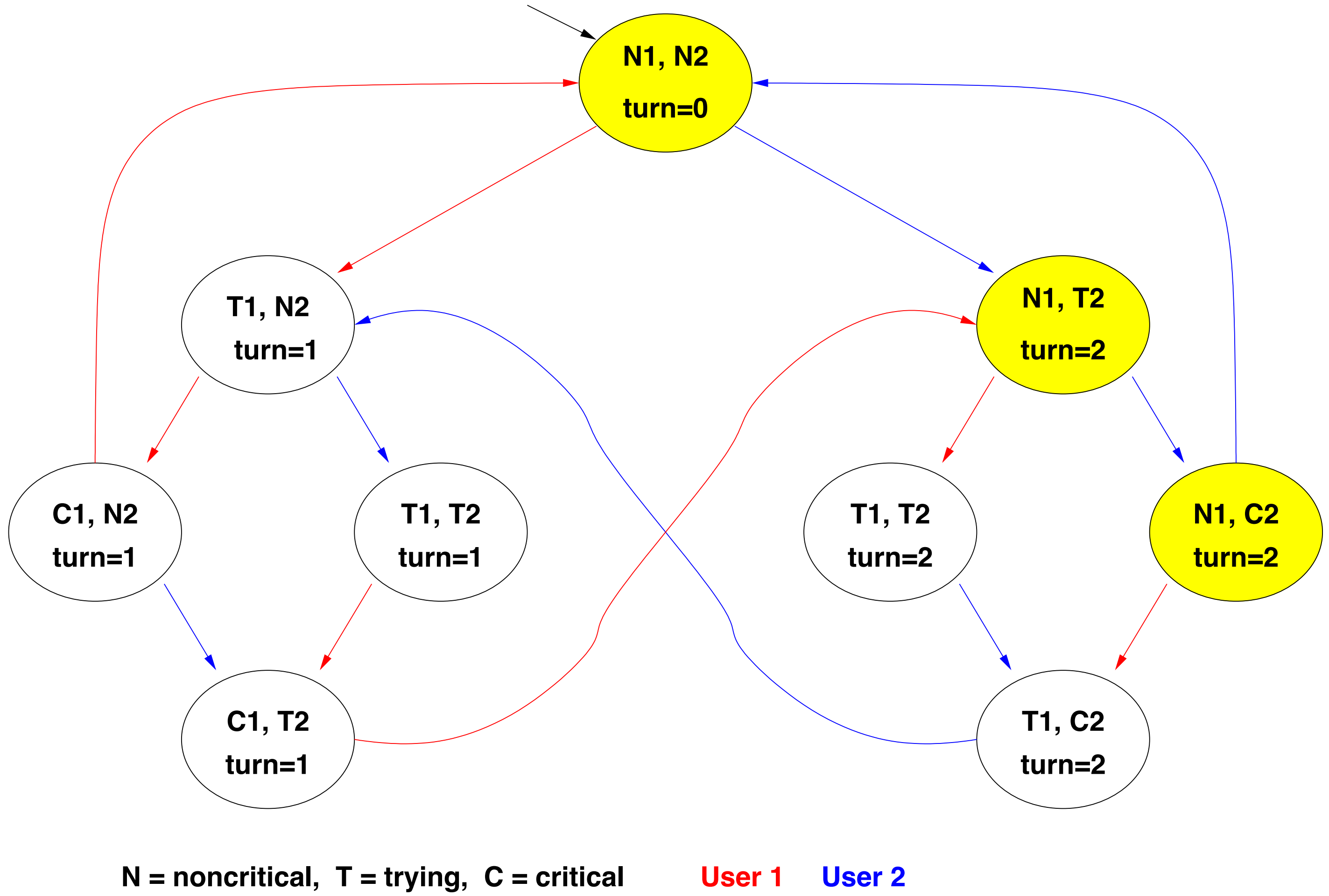
$[T_1]$:




$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1) \; ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1) \; ?$$

# Example 2: liveness

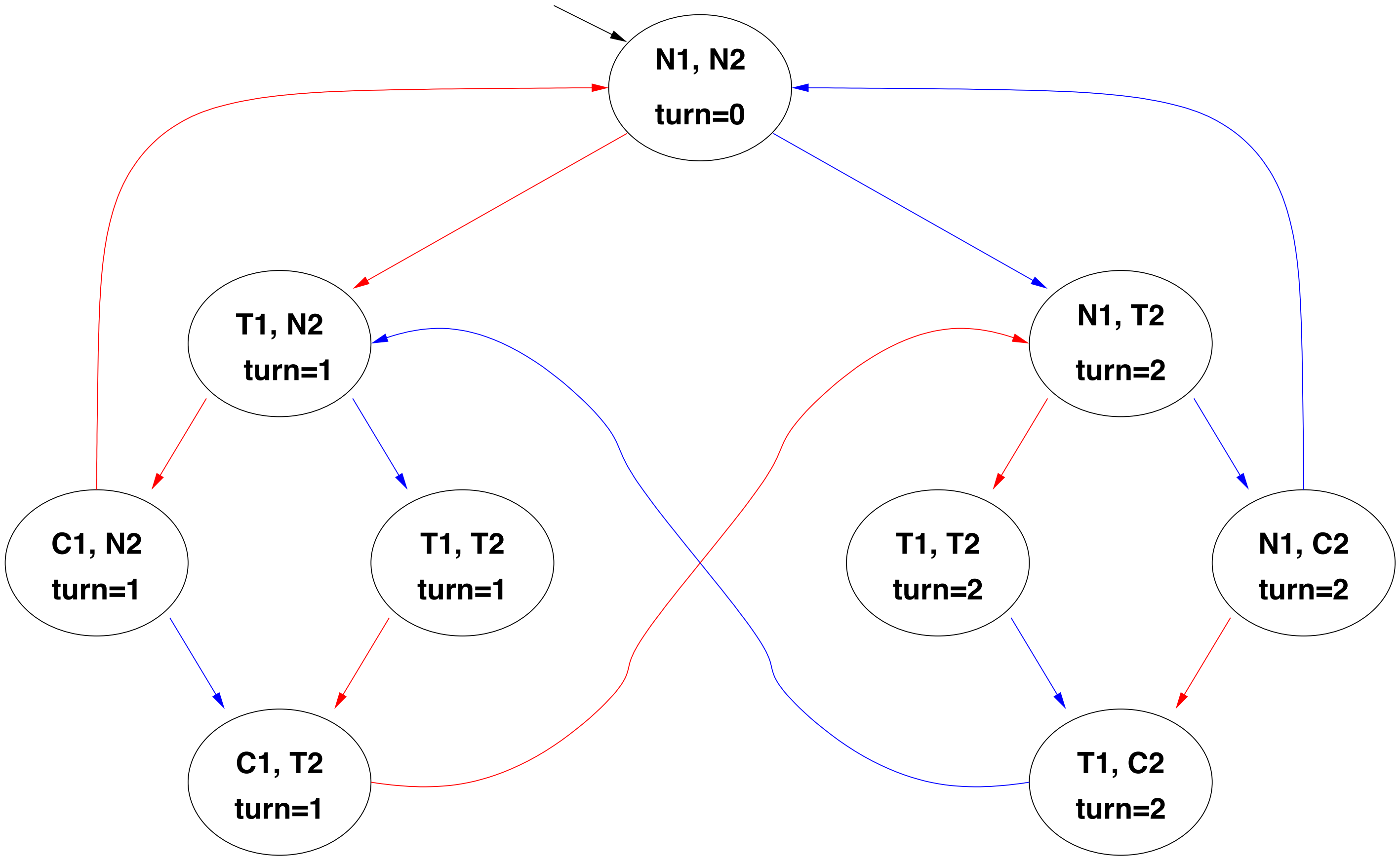
$[\mathbf{EG}\neg C_1]$, STEPS 0-4: (see previous example)




$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)\ ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)\ ?$$

# Example 2: liveness

$[T_1 \wedge \mathbf{EG}\neg C_1]$ :

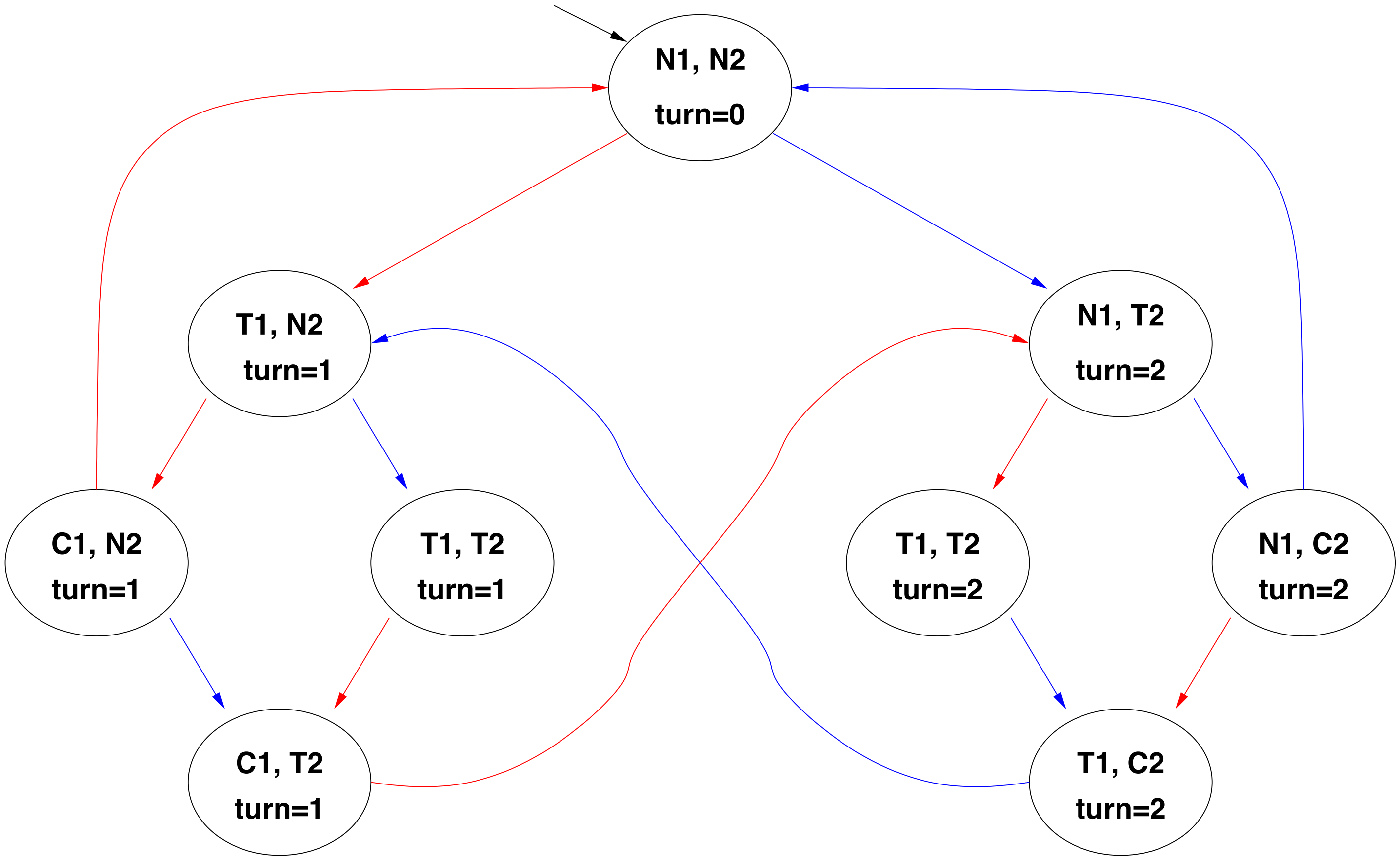



N = noncritical, T = trying, C = critical **User 1** **User 2**

$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)\ ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)\ ?$$

# Example 2: liveness

$[\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)]$ :



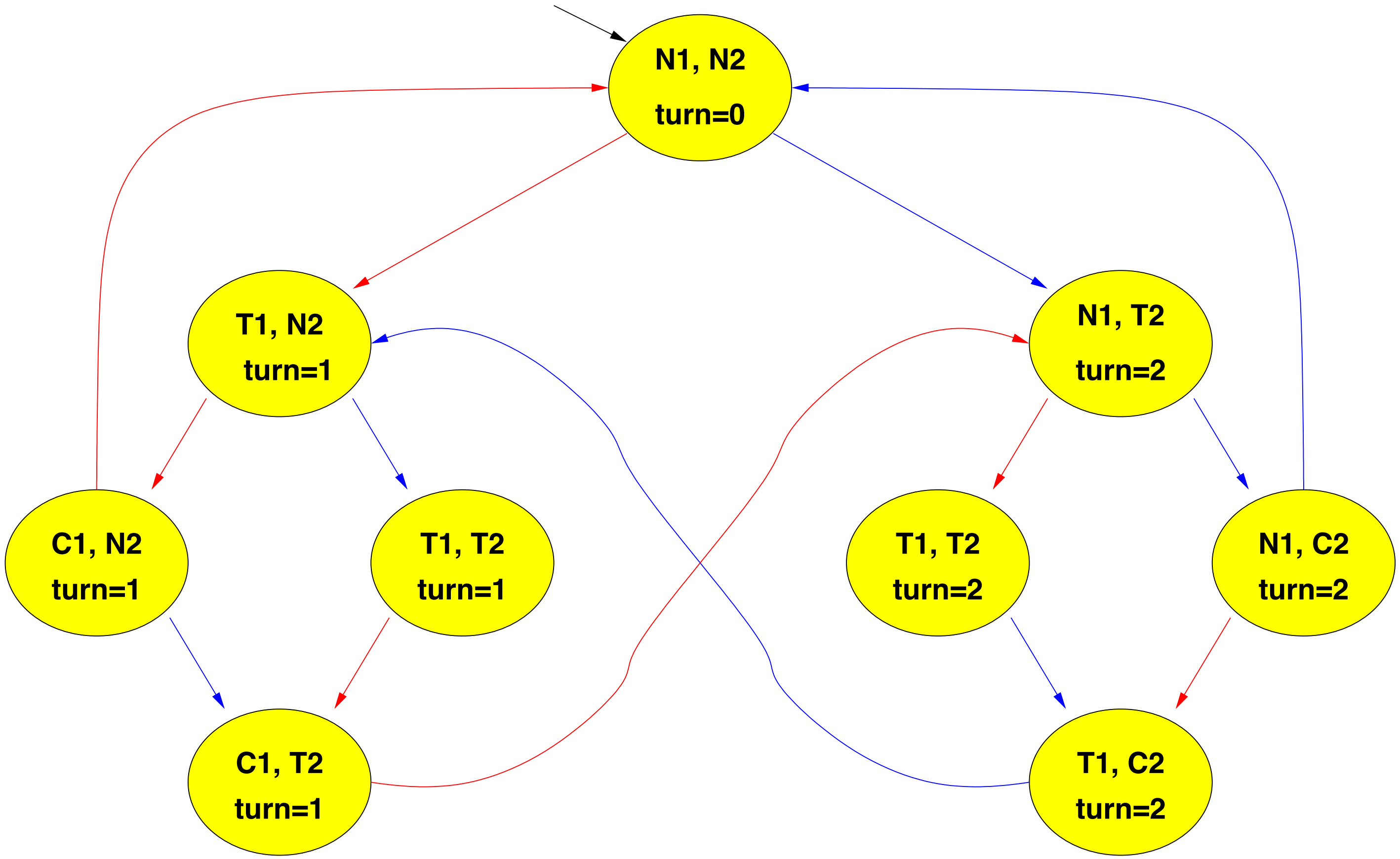

N = noncritical, T = trying, C = critical     User 1     User 2

$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)$ ? $\Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)$ ?

# Example 2: liveness

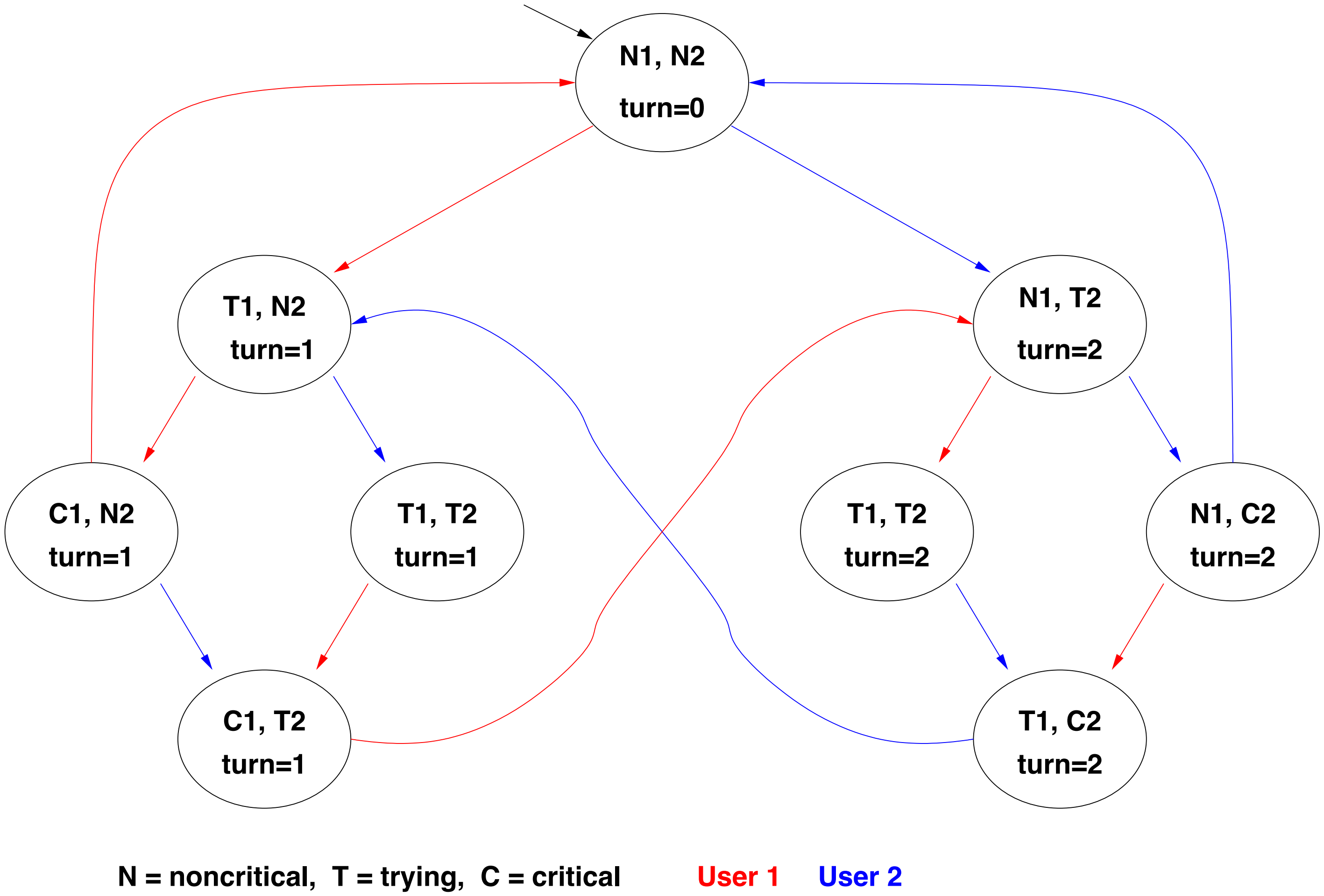
$[\neg \mathbf{EF}(T_1 \wedge \mathbf{EG} \neg C_1)]$ :




N = noncritical, T = trying, C = critical User 1 User 2

$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF} C_1)$ ? $\Longrightarrow M \models \neg \mathbf{EF}(T_1 \wedge \mathbf{EG} \neg C_1)$ ? YES!

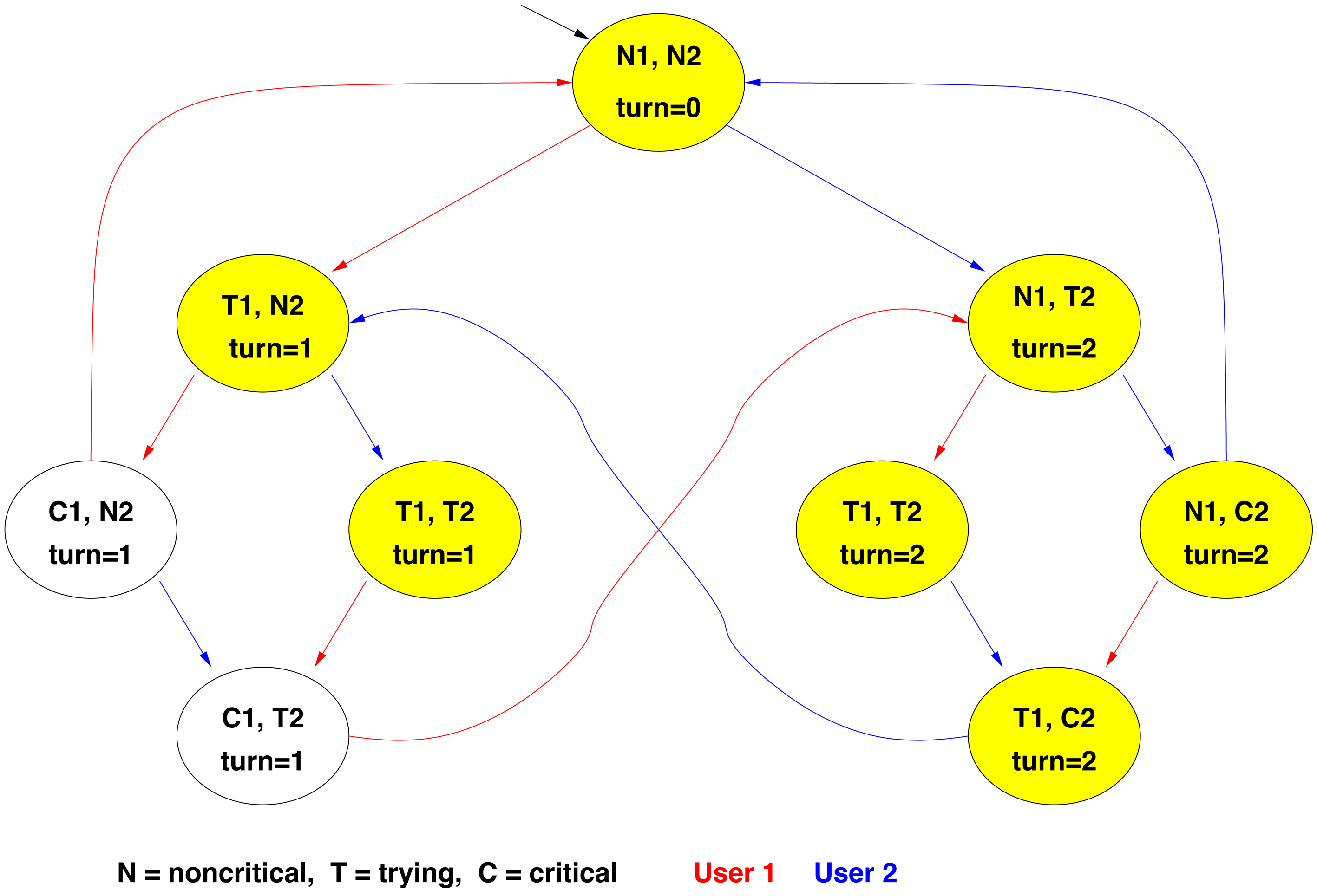
# Example 1: fairness




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

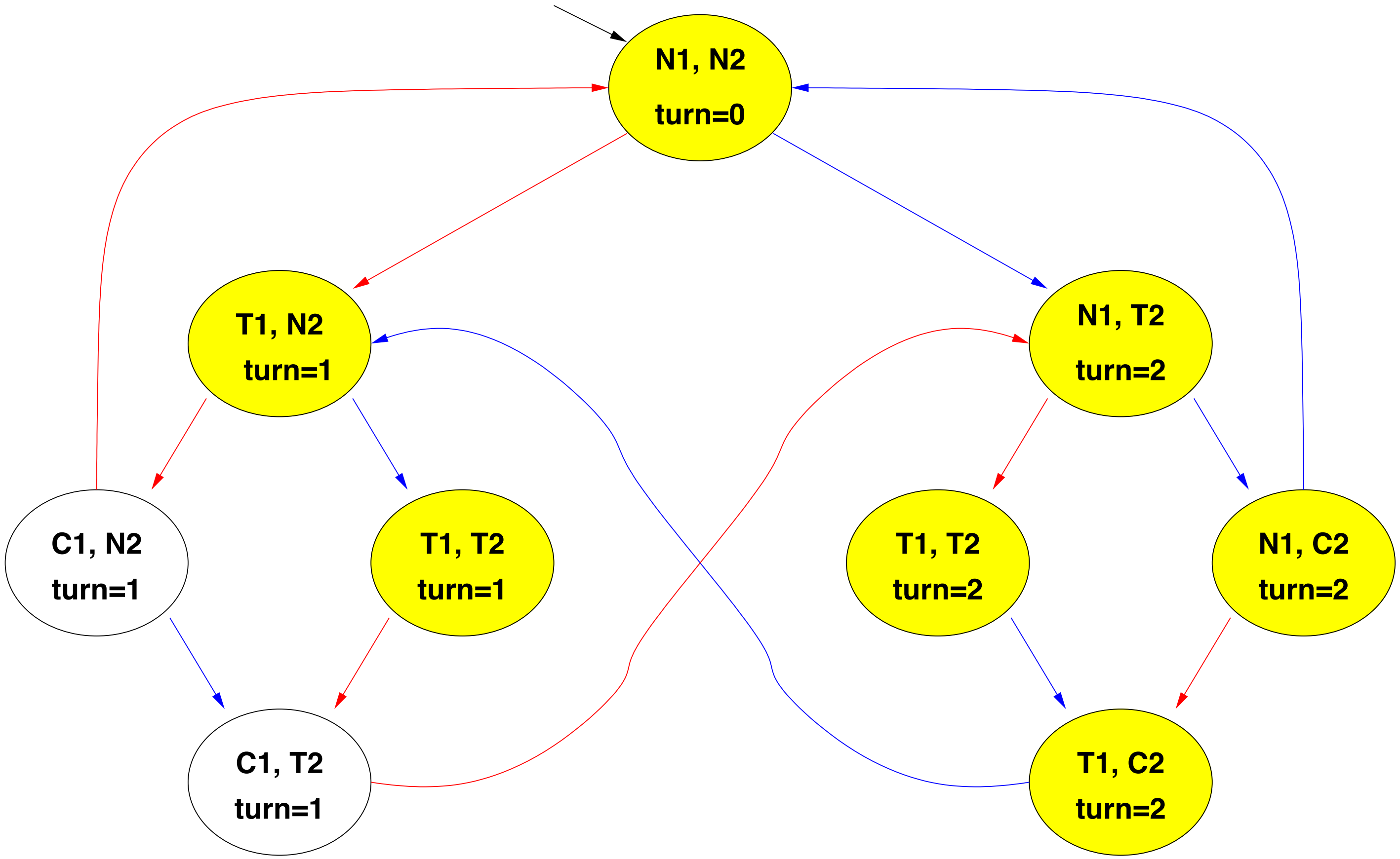
$[\neg C_1]$




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

$[\mathbf{EG}\neg C_1]$, step 0:

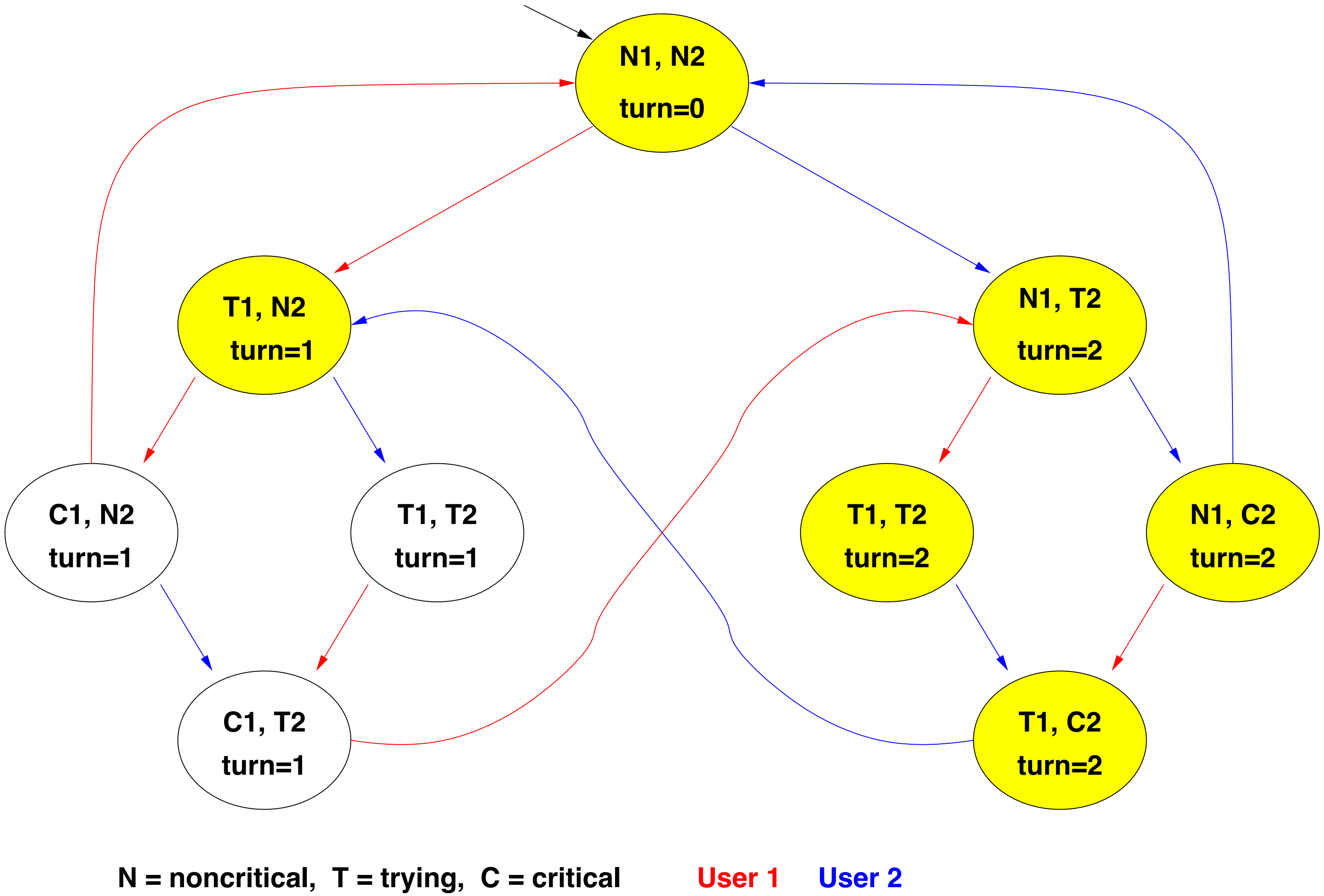



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

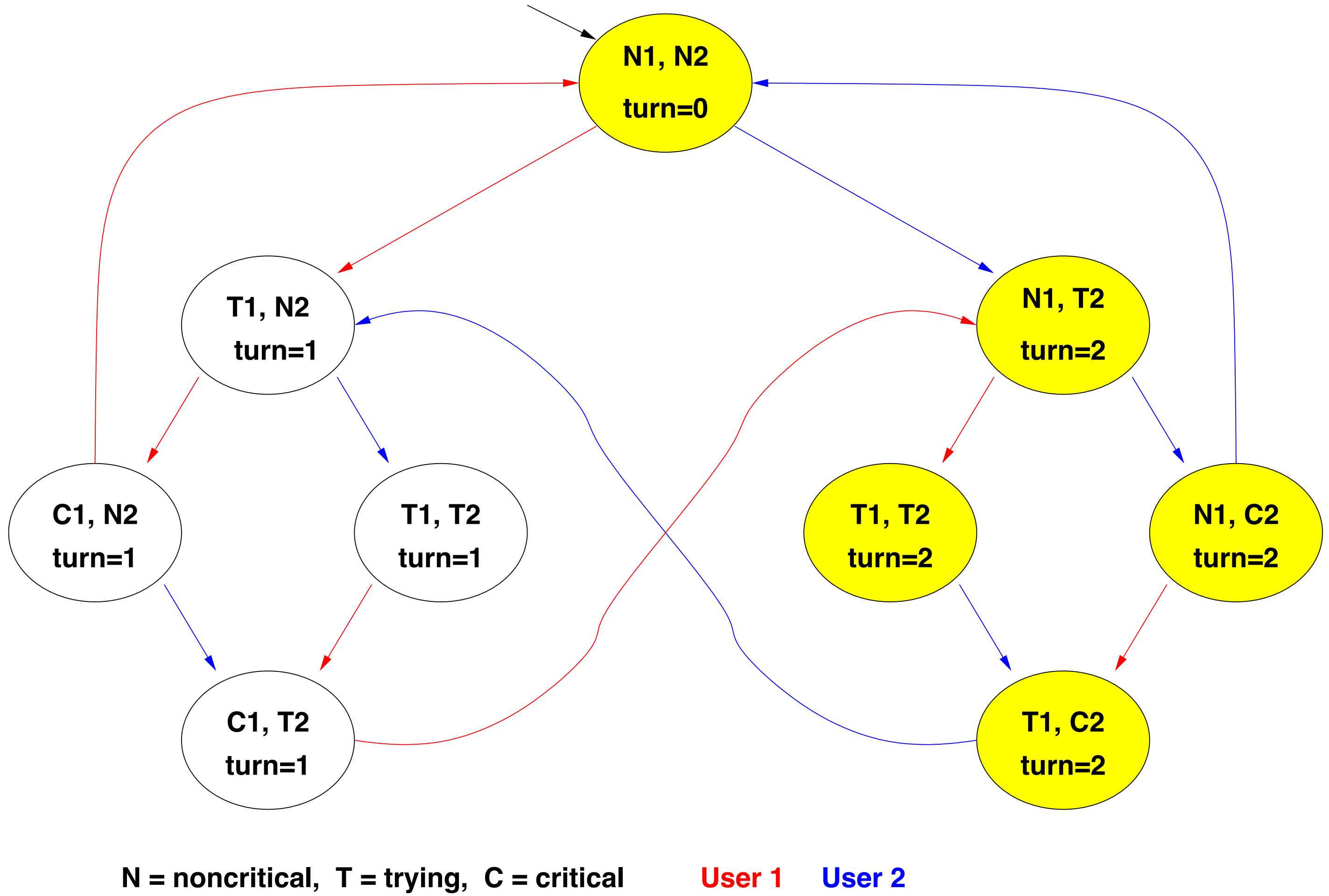
$[\mathbf{EG}\neg C_1]$, step 1:




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

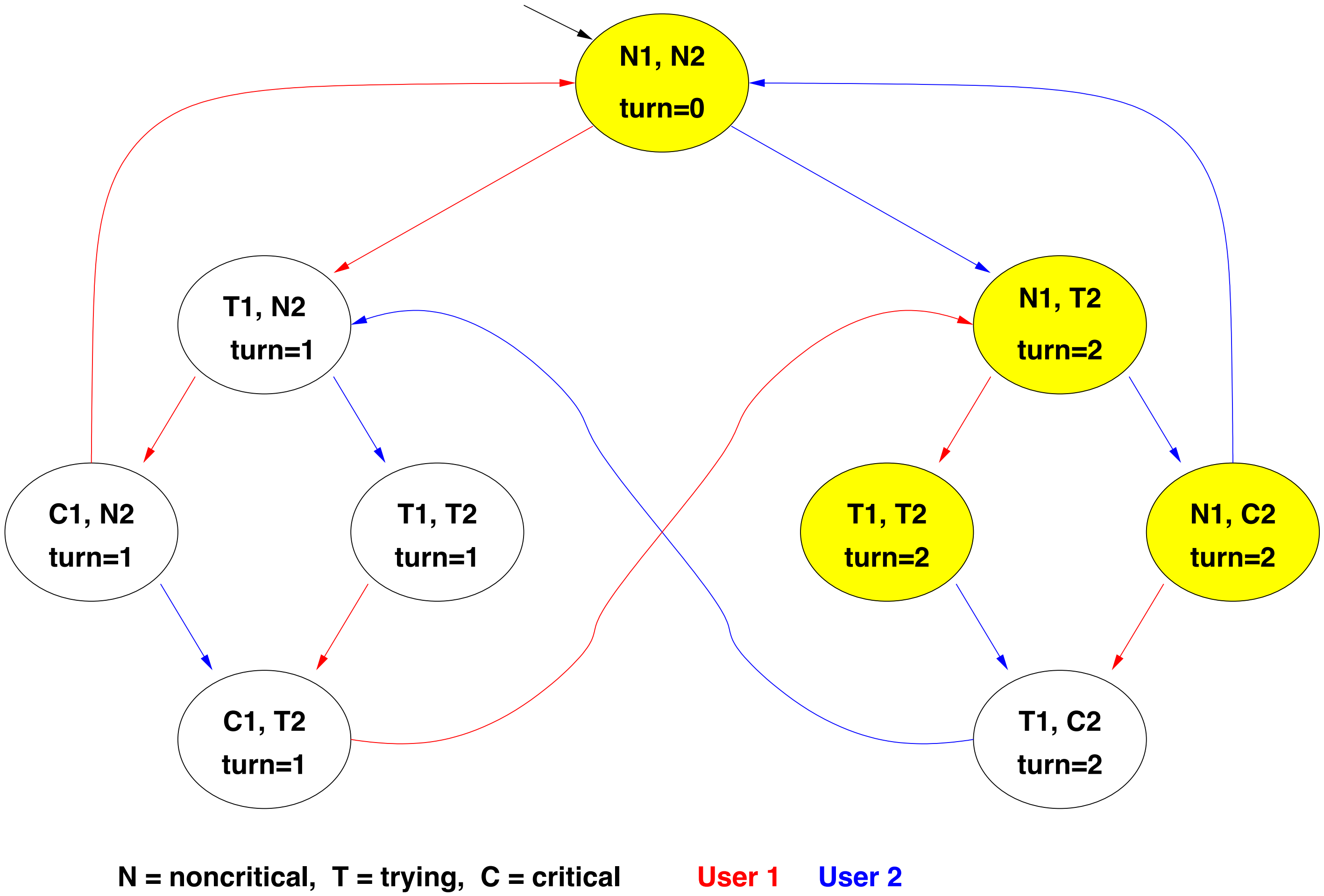
$[\mathbf{EG}\neg C_1]$, step 2:




$$M \models \mathbf{AGAF}C_1\ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1\ ?$$

# Example 1: fairness

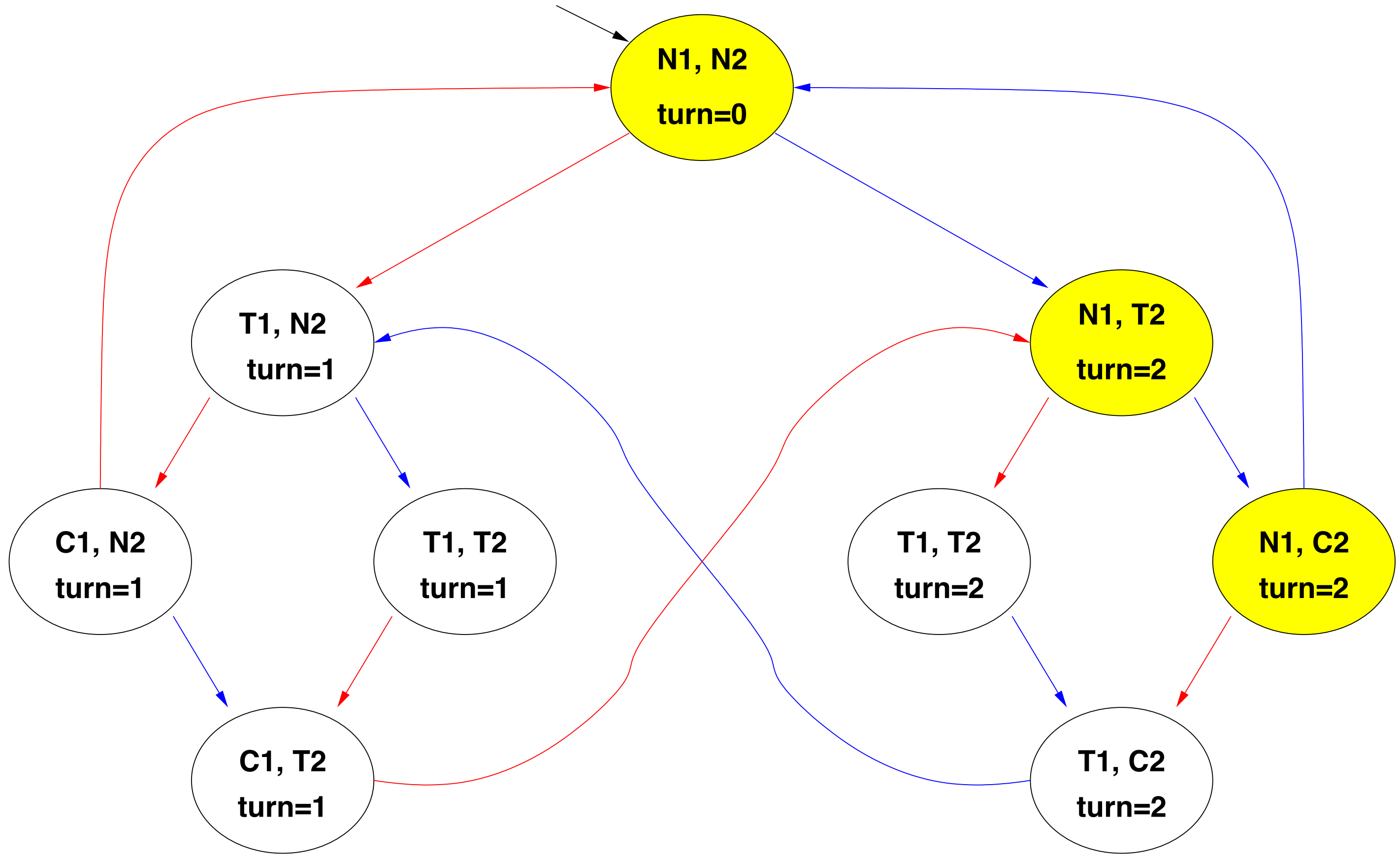
$[\mathbf{EG}\neg C_1]$, step 3:




$M \models \mathbf{AGAF}C_1\ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1\ ?$

# Example 1: fairness

$[\mathbf{EG}\neg C_1]$, step 4:

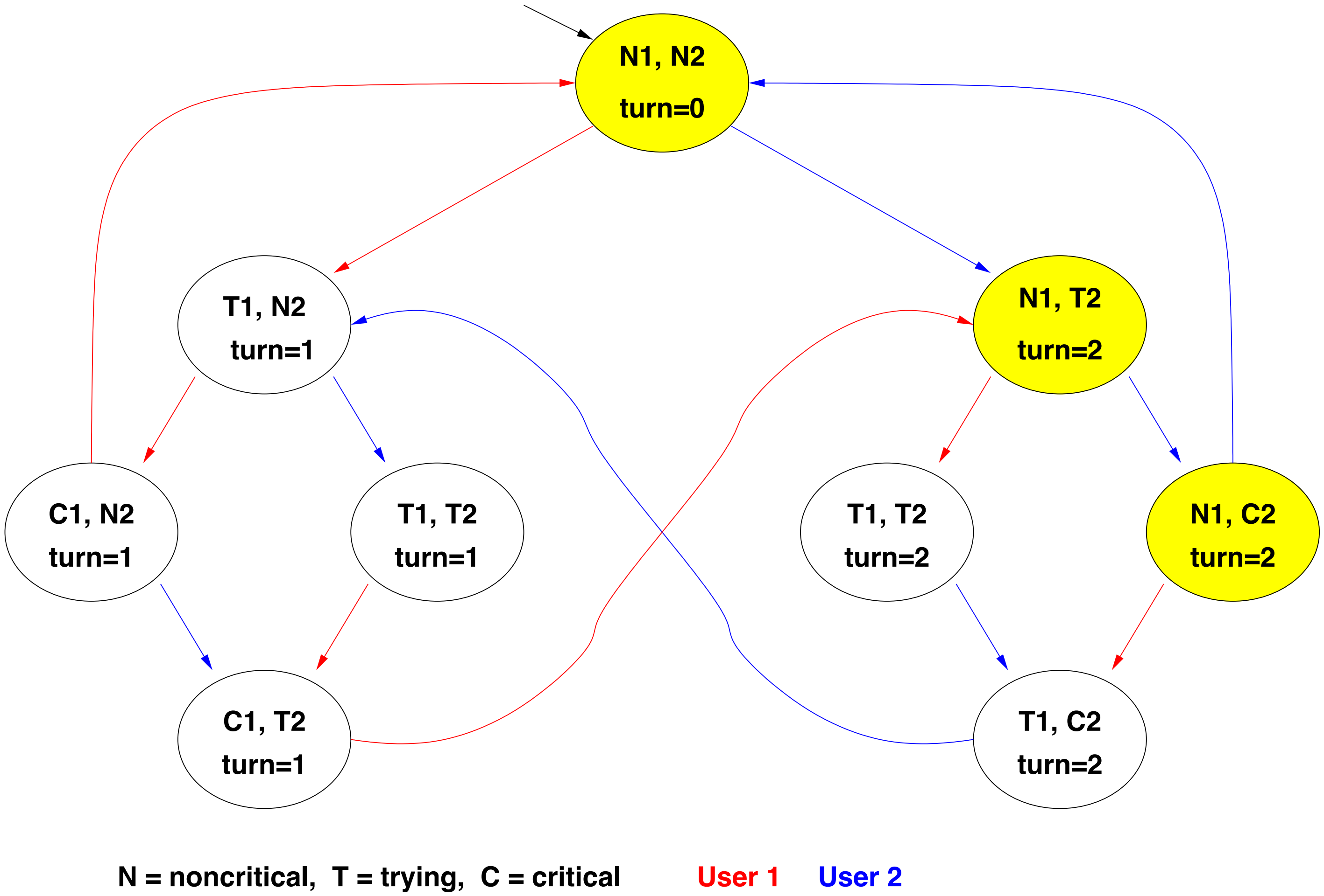



N = noncritical, T = trying, C = critical User 1 User 2

$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$

# Example 1: fairness

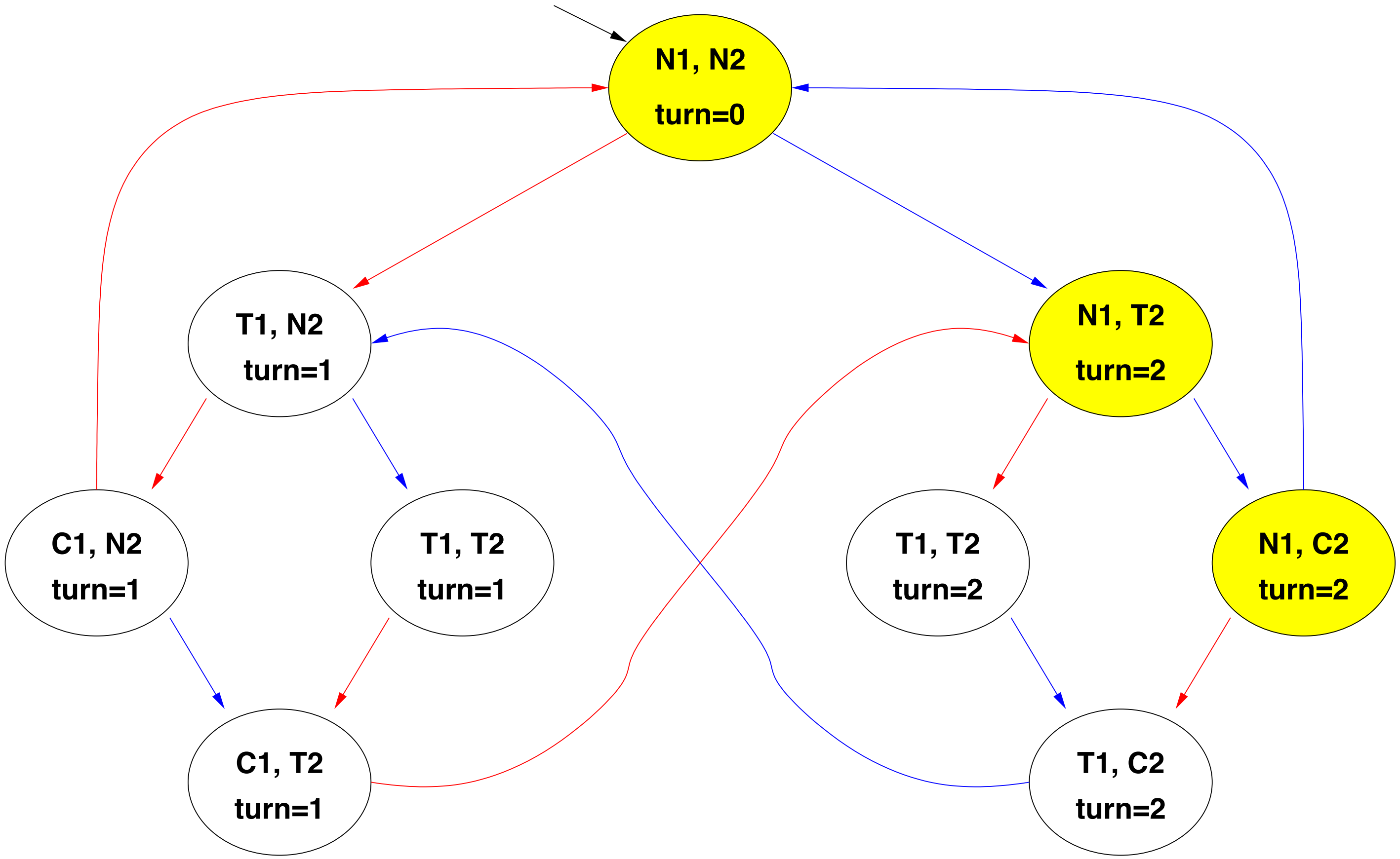
$[\mathbf{EG}\neg C_1]$, FIXPOINT!




$M \models \mathbf{AGAF}C_1$ ? $\Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1$ ?

# Example 1: fairness

## $[\mathbf{EFEG}\neg C_1]$, STEP 0

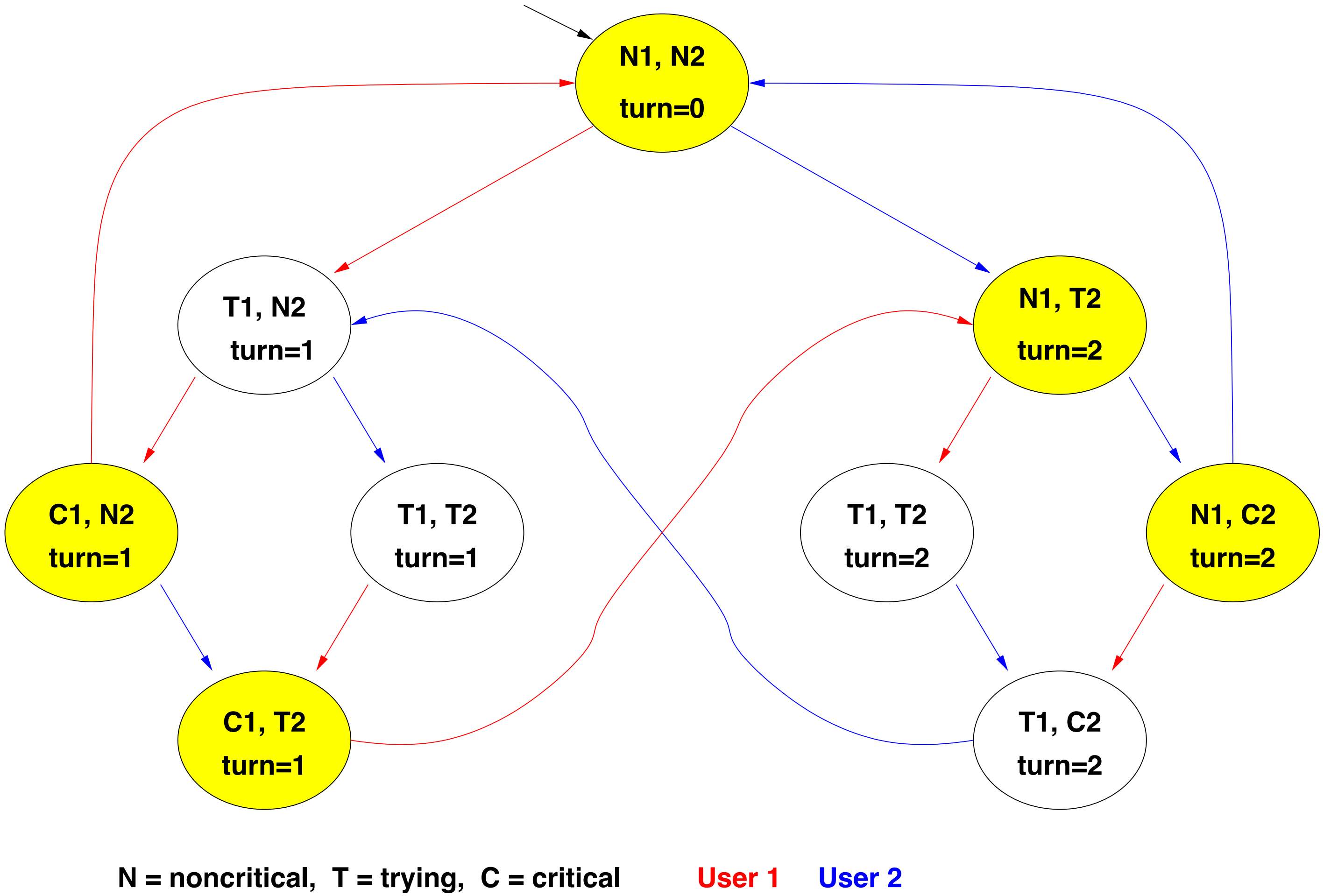



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AGAF}C_1\ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1\ ?$$

# Example 1: fairness

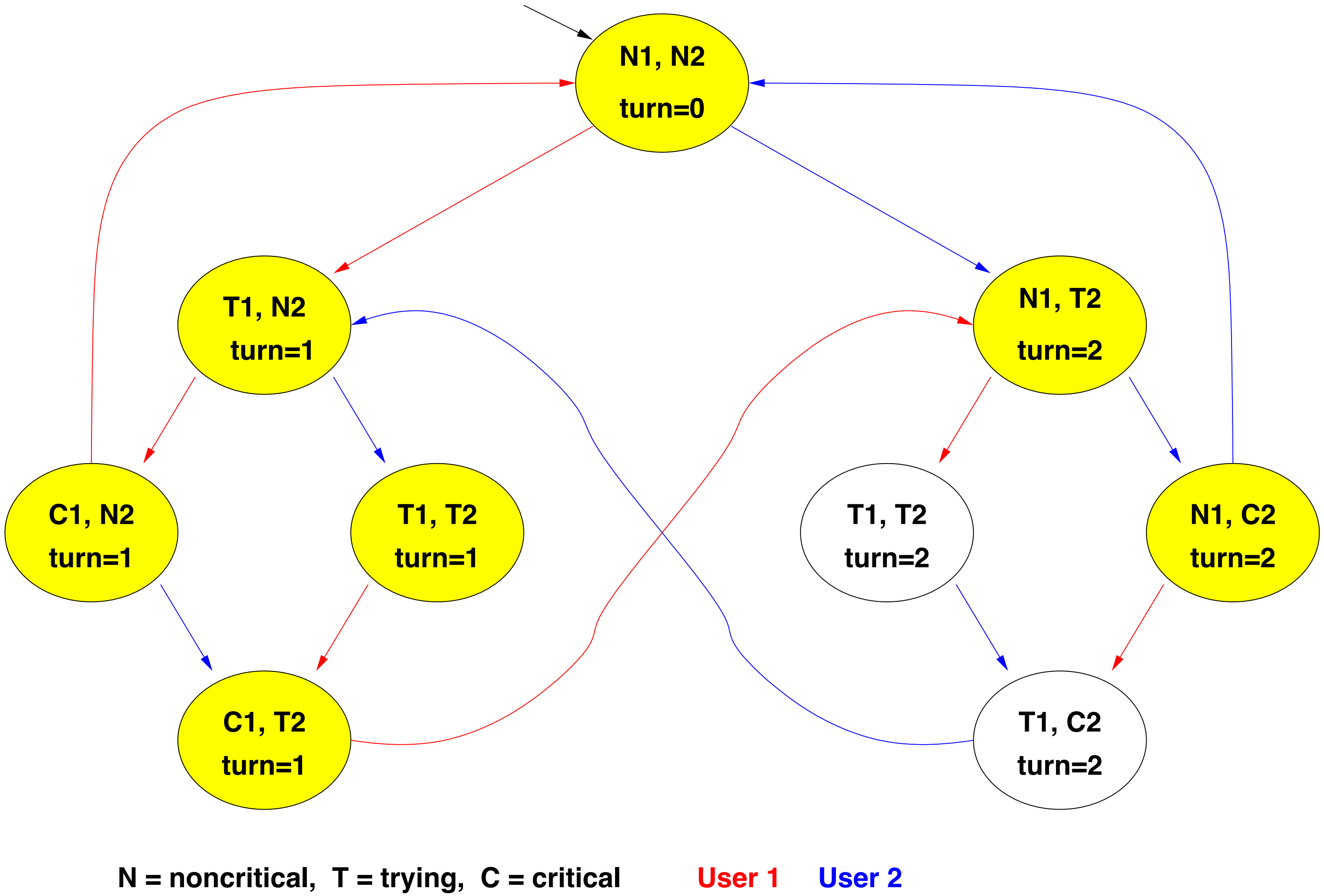
## $[\mathbf{EFEG}\neg C_1]$, STEP 1




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

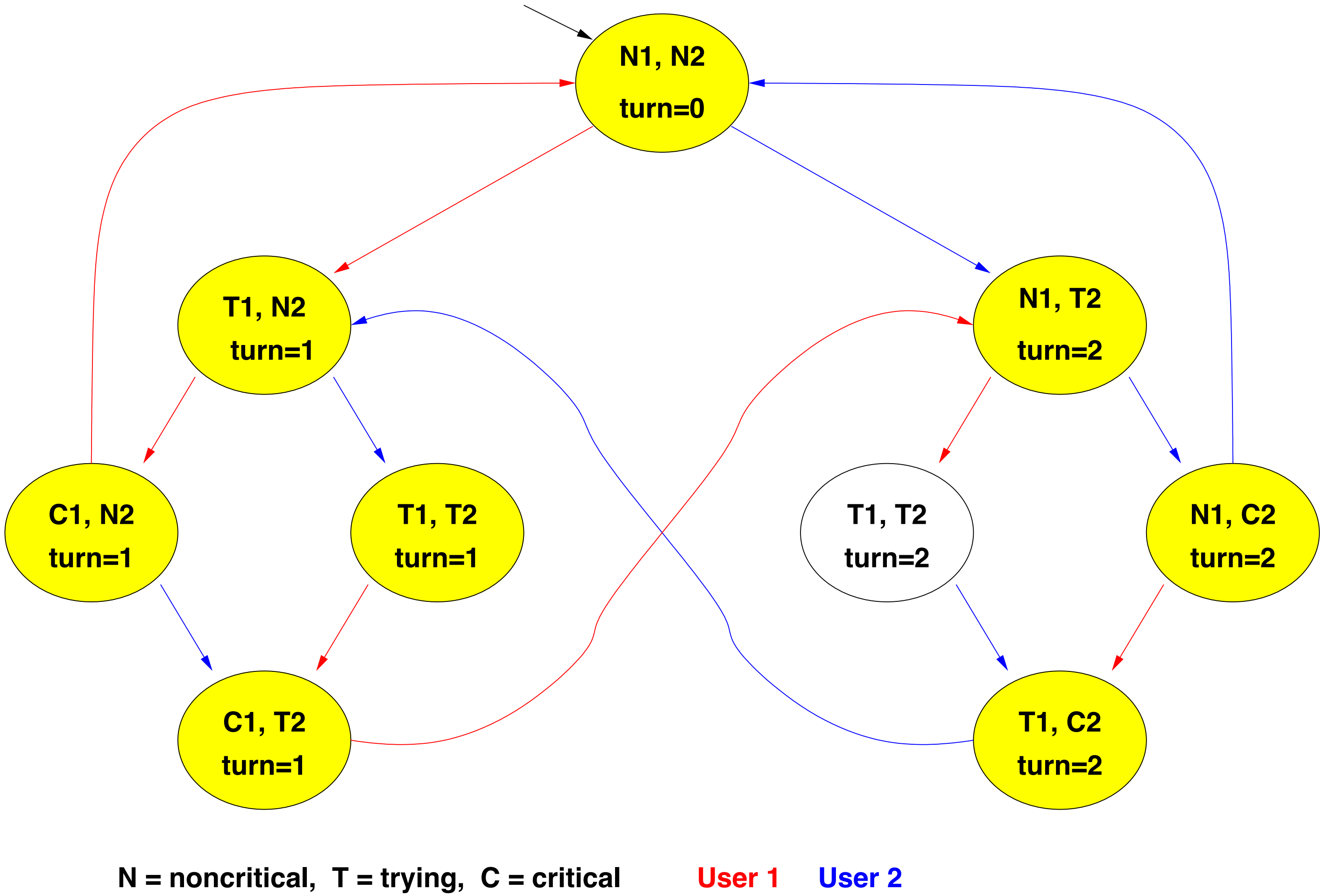
## $[\mathbf{EFEG}\neg C_1]$, STEP 2




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

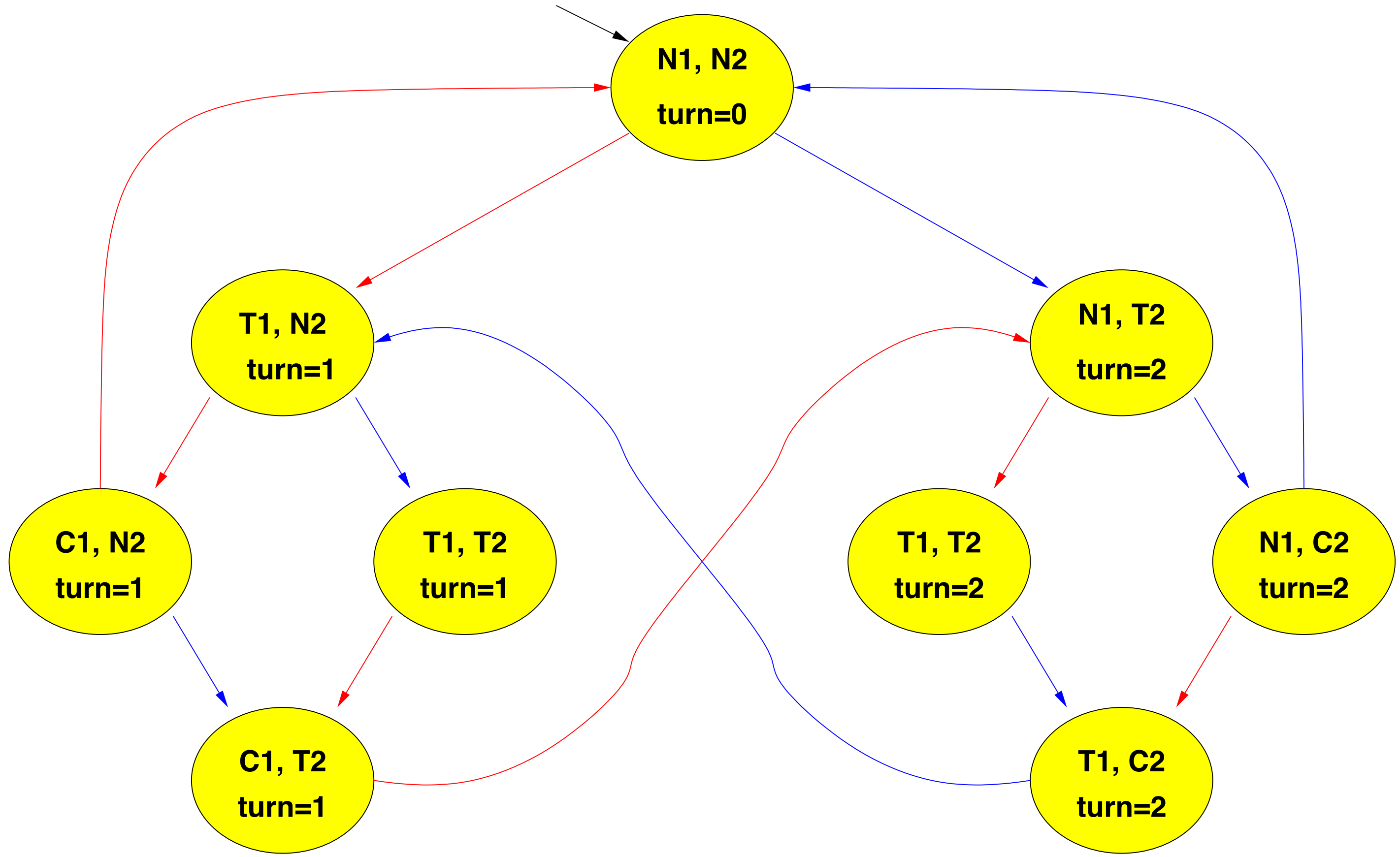
## $[\mathbf{EFEG}\neg C_1]$, STEP 3




$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

## $[\mathbf{EFEG}\neg C_1]$, STEP 4

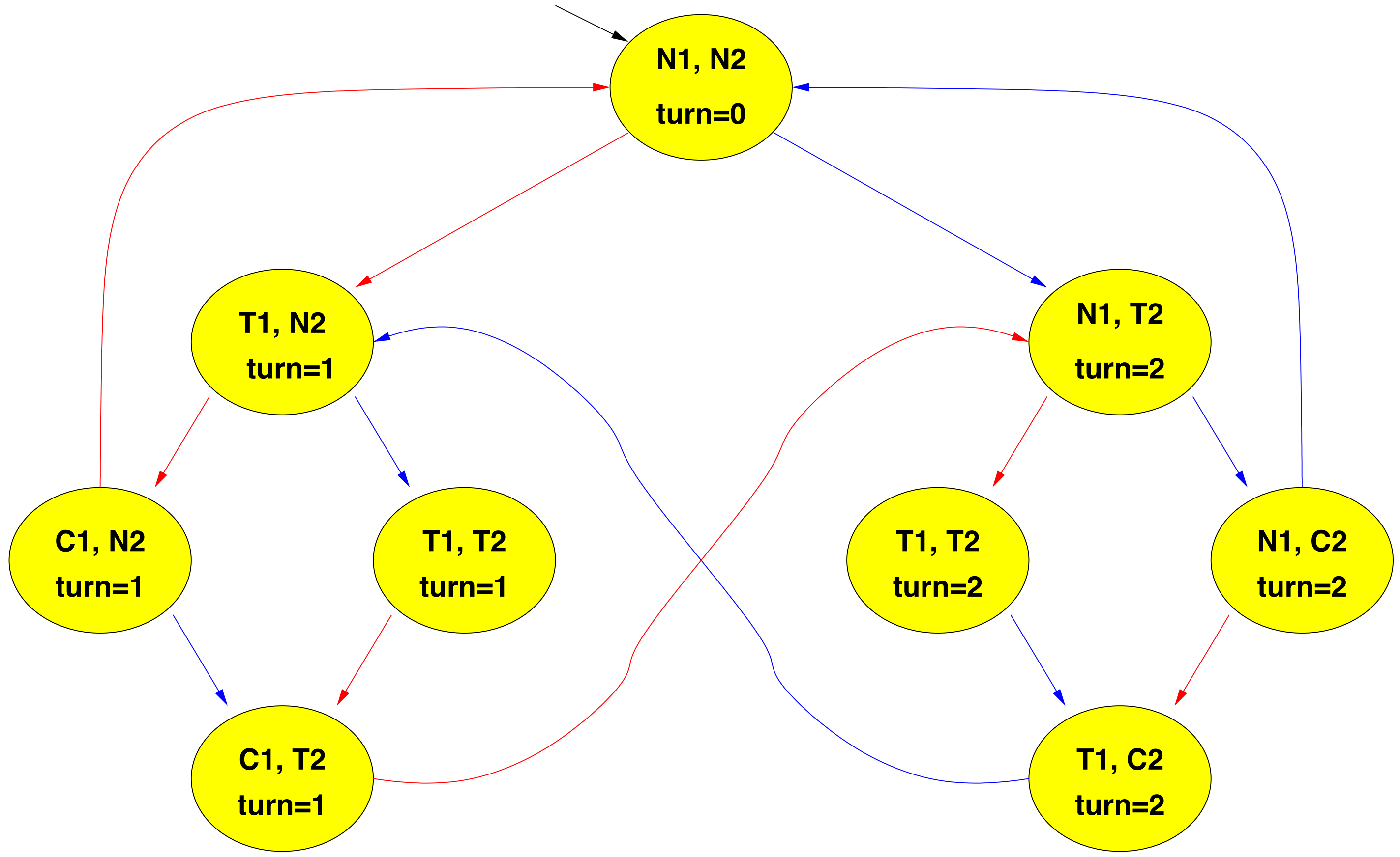



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFEG}\neg C_1 \ ?$$

# Example 1: fairness

## $[\mathbf{EFG}\neg C_1]$, FIXPOINT!



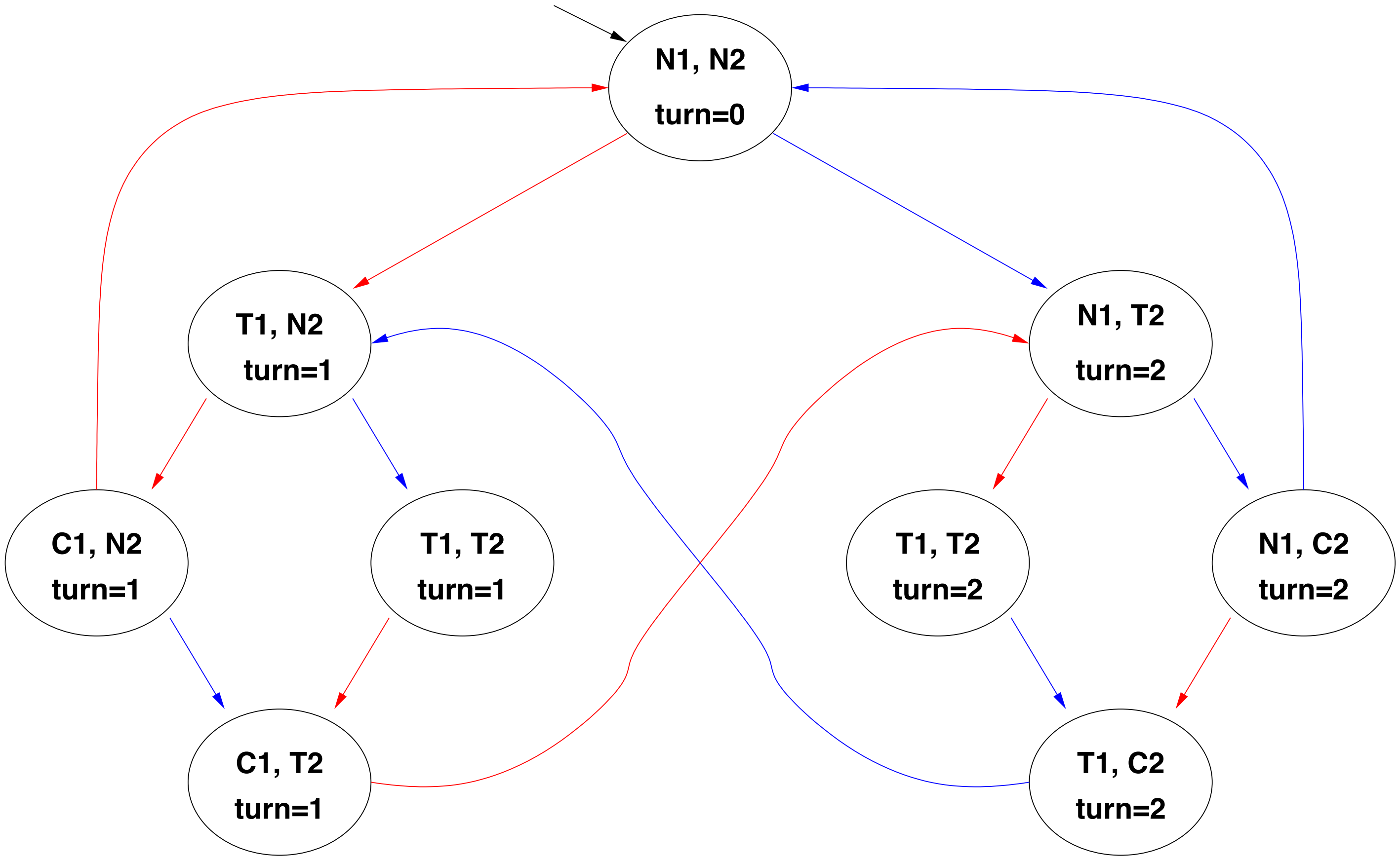

N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AGAF}C_1 \ ? \Longrightarrow M \models \neg\mathbf{EFG}\neg C_1 \ ?$$

# Example 1: fairness

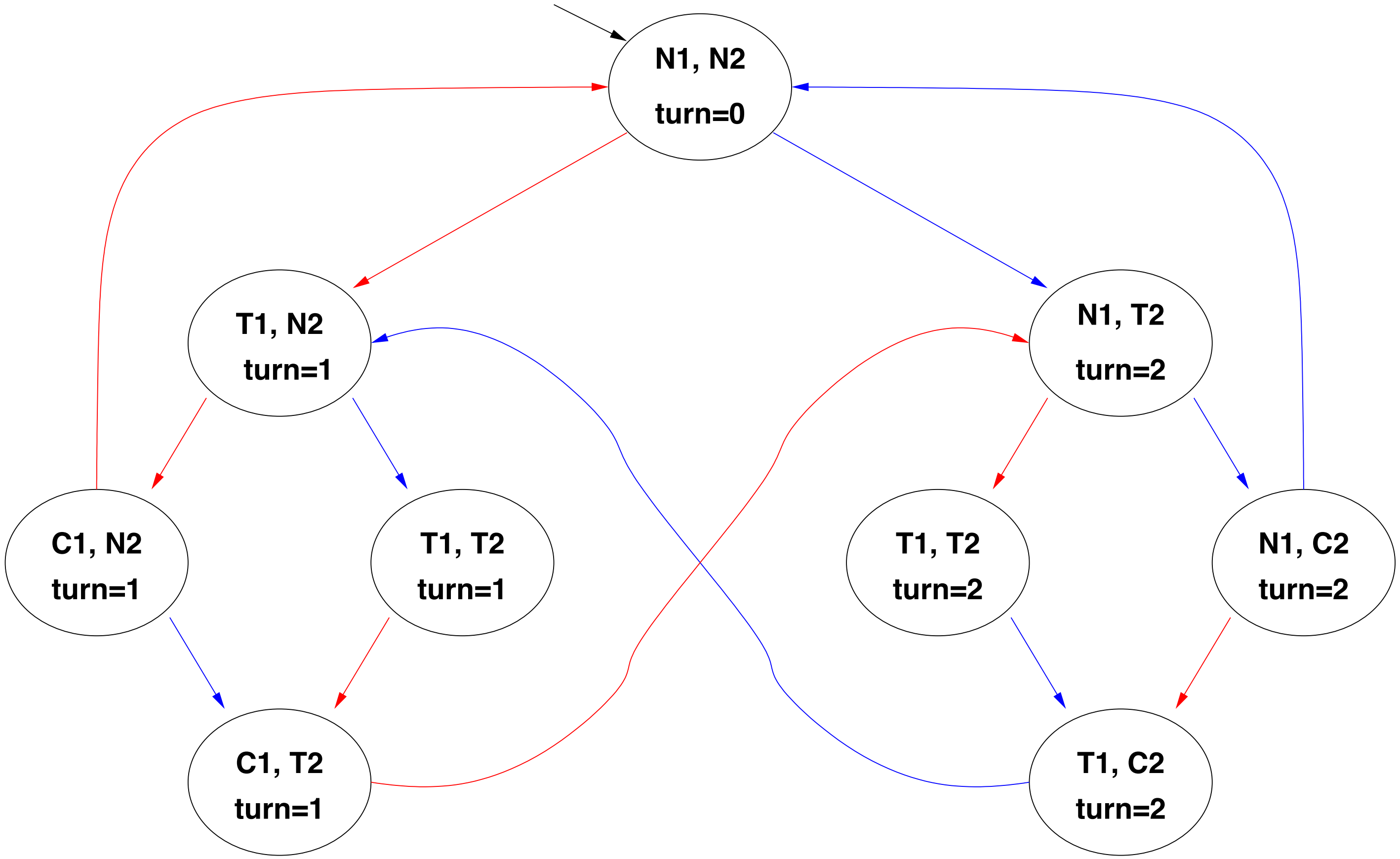
$[\neg \mathbf{EFEG} \neg C_1]$




N = noncritical, T = trying, C = critical     User 1    User 2

$M \models \mathbf{AGAF} C_1 \ ? \Longrightarrow M \models \neg \mathbf{EFEG} \neg C_1 \ ? \Longrightarrow$ NO!

# Example 2: liveness

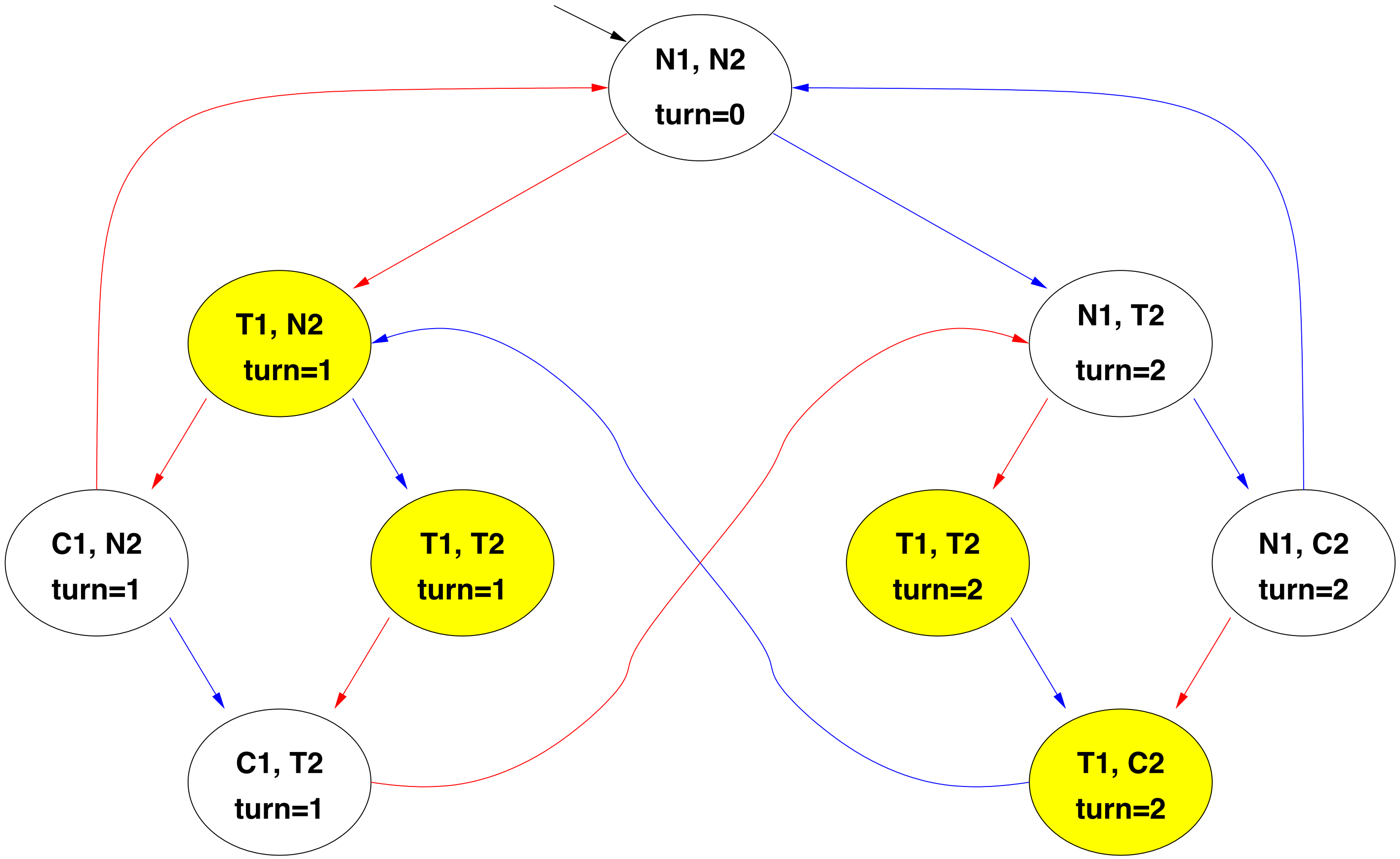



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)\ ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)\ ?$$

# Example 2: liveness

$[T_1]$:

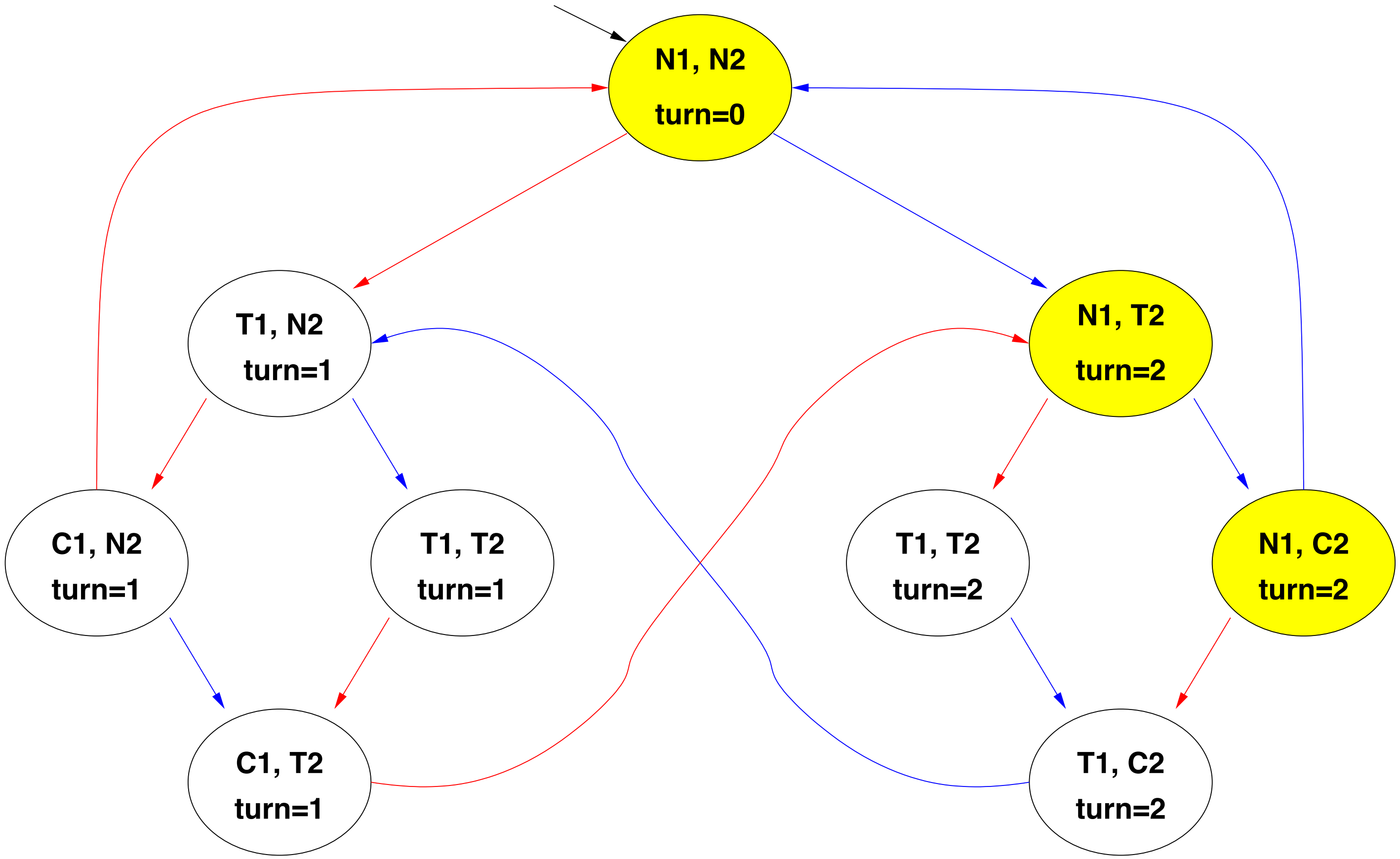



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)\ ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)\ ?$$

# Example 2: liveness

$[\mathbf{EG}\neg C_1]$, STEPS 0-4: (see previous example)

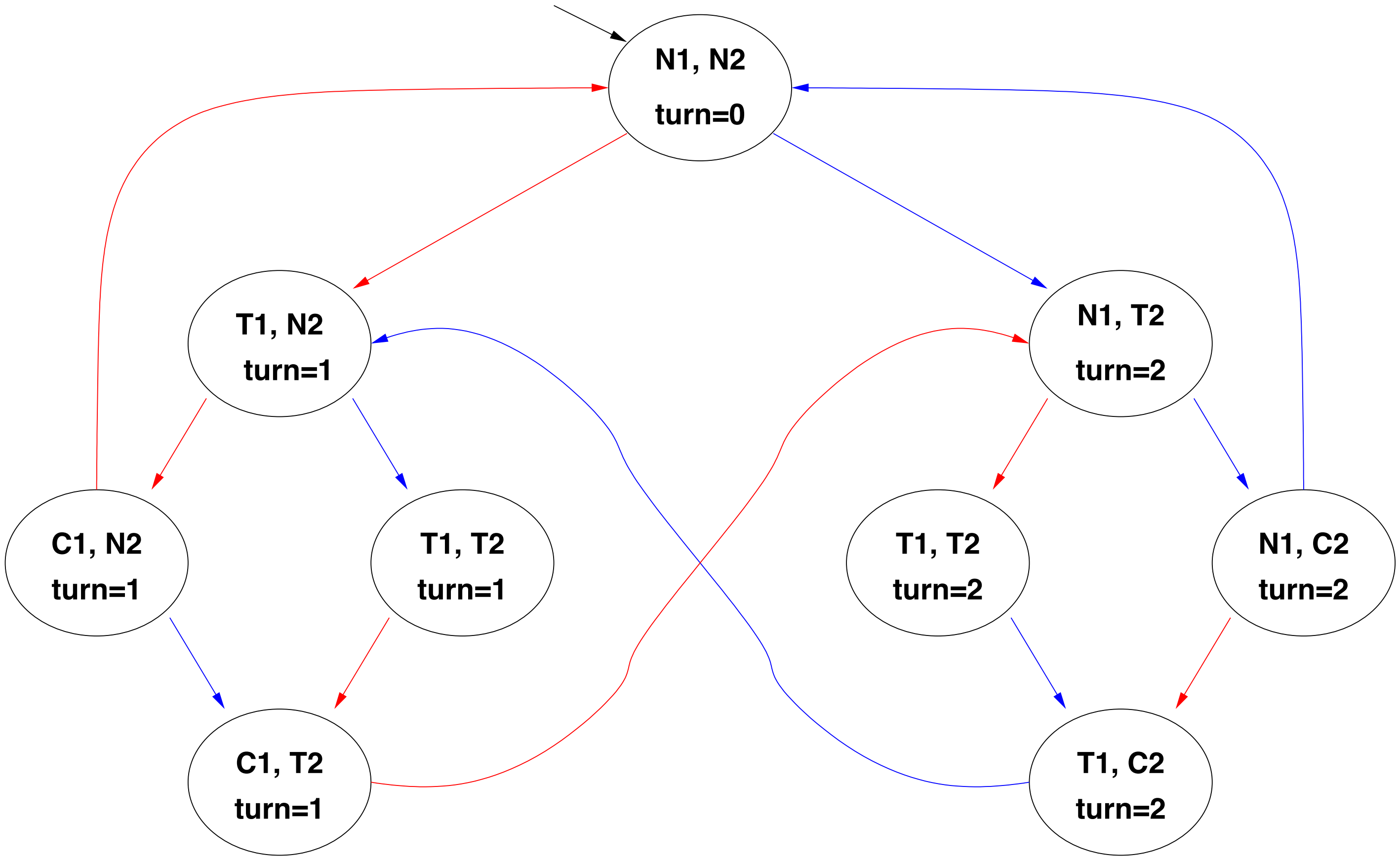



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)\ ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)\ ?$$

# Example 2: liveness

$[T_1 \wedge \mathbf{EG}\neg C_1]$ :

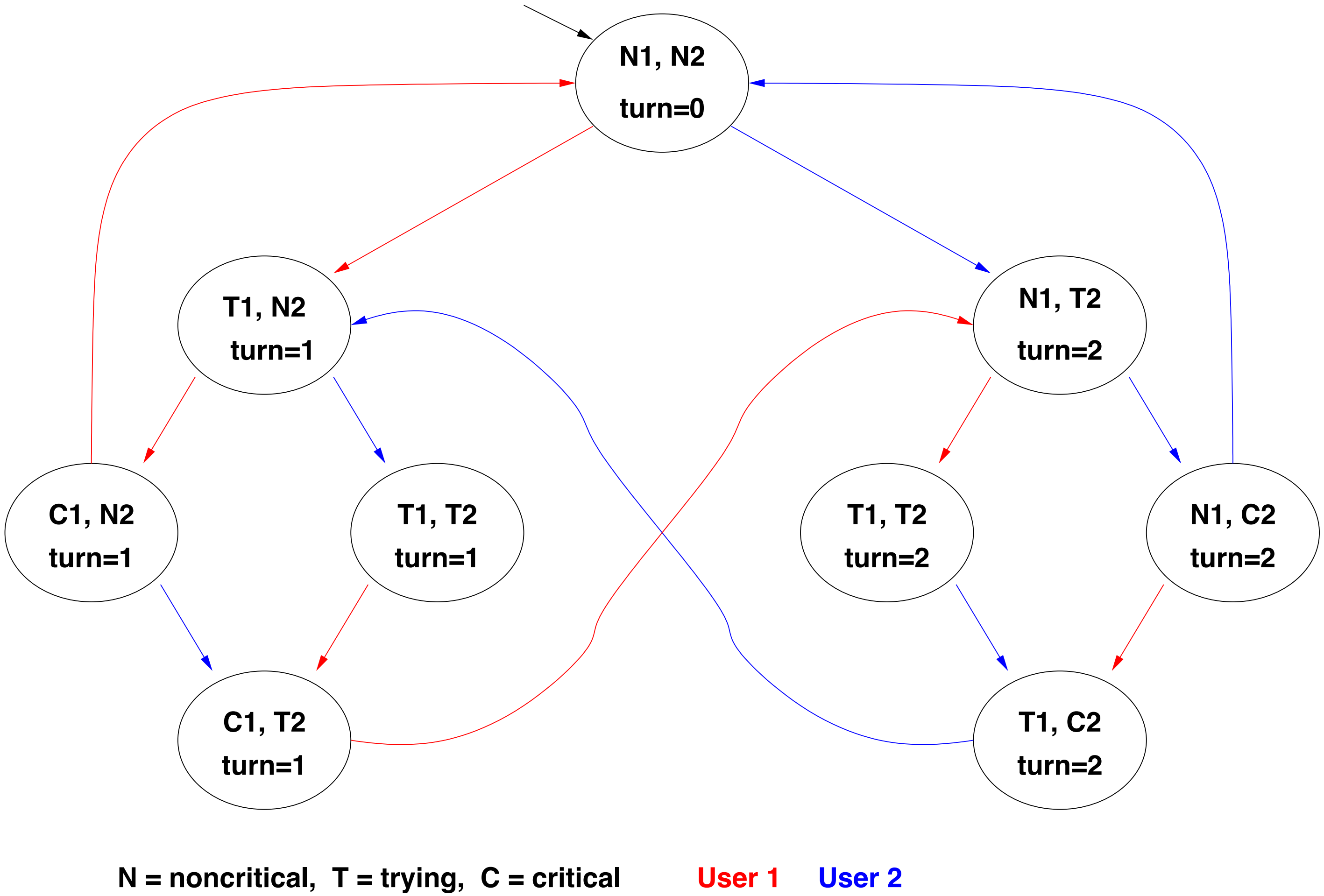



N = noncritical, T = trying, C = critical User 1 User 2

$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)\ ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)\ ?$$

# Example 2: liveness

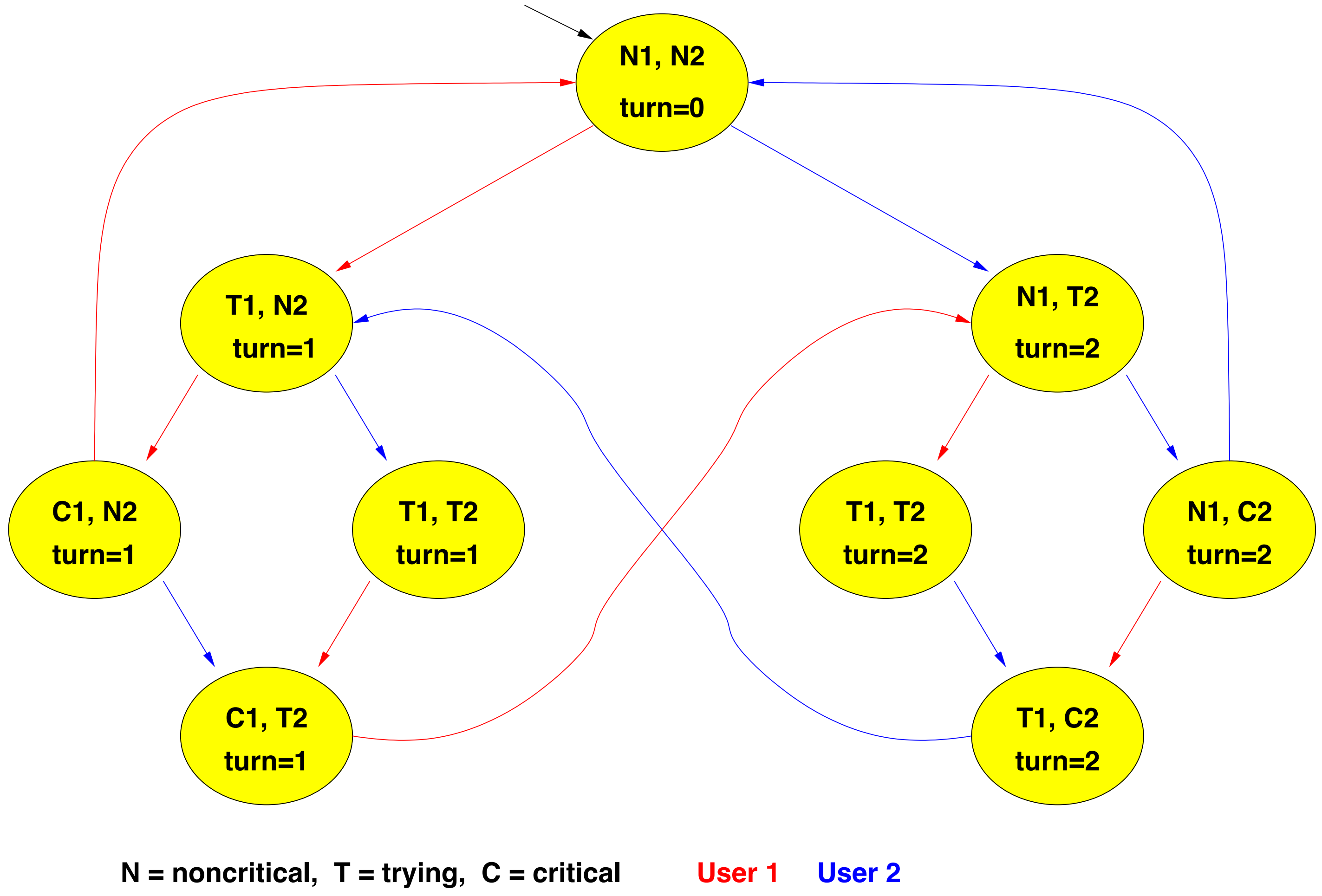
$[\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)]$ :




$$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)\ ? \Longrightarrow M \models \neg\mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)\ ?$$

# Example 2: liveness

$[\neg \mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)]$ :




$M \models \mathbf{AG}(T_1 \rightarrow \mathbf{AF}C_1)$ ? $\Longrightarrow M \models \neg \mathbf{EF}(T_1 \wedge \mathbf{EG}\neg C_1)$ ? YES!