Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Semantics of CTL

what we can sav

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

October 21, 2015

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Semantics of CTL what we can sav

- In previous chapters...
- Motivation and Intuition
- How to communicate
 - Syntax of CTL
 - Semantics of CTI
- 4 Some examples of what we can say
- More about semantics
 - Equivalences

Previously on Temporal Logic Week... Temporal Logic

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...

Motivation and Intuition How to communicate Syntax of CTI

Syntax of CTL Semantics of CTL

Some examples of what we can say

More about semantics Equivalences



Motivation

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous

chapters...

Motivation and

How to communicate Syntax of CTL Semantics of

Semantics of CTL
Some examples of what we can say

More about semantics ■ Needing of uncertainty;

Motivation

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous

chapters...

Motivation and Intuition

How to communicate Syntax of CT Semantics of CTL

Some examples of what we can sav

More about semantics Equivalences

- Needing of uncertainty;
- Different paths of the future;

Intuition

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...

Motivation and

How to communicate Syntax of CT

Syntax of CTL Semantics of CTL Some examples

what we can say

More about
semantics
Equivalences

In Computation Tree Logic (CTL) the model of time is a tree-like structure. This way, we cannot use Linear Temporal Logic (LTL) to express the existence of a certain path of time in which some event occurs.

Syntax Definition

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Syntax of CTL

Semantics of CTL

what we can sav

The syntax of CTL consists on the syntax of temporal logic plus some path operators. The class of formulas can be defined in Backus-Naur form. If ϕ is a formula:

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...

Motivation an

How to

Syntax of CTL Semantics of CTL

CTL Some examples of

what we can say

More about semantics
Equivalences

The syntax of CTL consists on the syntax of temporal logic plus some path operators. The class of formulas can be defined in Backus-Naur form. If ϕ is a formula:

$$\begin{split} \phi ::= \bot \mid \top \mid \rho \mid \neg \phi \mid \phi \land \phi \mid \phi \lor \phi \mid \phi \to \phi \mid \mathsf{AX}\phi \mid \mathsf{EX}\phi \mid \\ \mathsf{AF}\phi \mid \mathsf{EF}\phi \mid \mathsf{AG}\phi \mid \mathsf{EG}\phi \mid \mathsf{A}[\phi \mathsf{U}\phi] \mid \mathsf{E}[\phi \mathsf{U}\phi] \end{split}$$

Syntax Definition

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...

Motivation a Intuition

How to

Syntax of CTL Semantics of CTL

CTL Some examples of what we can sav

More about semantics Equivalences The syntax of CTL consists on the syntax of temporal logic plus some path operators. The class of formulas can be defined in Backus-Naur form. If ϕ is a formula:

$$\phi ::= \bot \mid \top \mid \rho \mid \neg \phi \mid \phi \land \phi \mid \phi \lor \phi \mid \phi \rightarrow \phi \mid \mathsf{AX}\phi \mid \mathsf{EX}\phi \mid$$

$$\mathsf{AF}\phi \mid \mathsf{EF}\phi \mid \mathsf{AG}\phi \mid \mathsf{EG}\phi \mid \mathsf{A}[\phi\mathsf{U}\phi] \mid \mathsf{E}[\phi\mathsf{U}\phi]$$

With p as a literal (atomic formula), AX, EX, AF, EF, AG e EG unary operators.

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Syntax of CTL

Semantics of CTL

what we can sav

The propositional operators: $\neg, \lor, \land, \rightarrow$ have the same meaning of in the propositional logic.

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...

Motivation and Intuition

communicate
Syntax of CTL

Syntax of CTL Semantics of CTL

More about

The propositional operators: \neg , \lor , \land , \rightarrow have the same meaning of in the propositional logic.

The temporal operators can be read (if φ is a formula) as follows:

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...
Motivation and

How to

Syntax of CTL Semantics of CTL

CTL Some examples o what we can say

More about semantics Equivalences The propositional operators: \neg , \lor , \land , \rightarrow have the same meaning of in the propositional logic.

The temporal operators can be read (if φ is a formula) as follows:

■ A φ : φ is true in all possible paths;

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and

Syntax of CTL Semantics of CTL

what we can sav

The propositional operators: $\neg, \lor, \land, \rightarrow$ have the same meaning of in the propositional logic.

The temporal operators can be read (if φ is a formula) as follows:

- \blacksquare A φ : φ is true in all possible paths;
- \blacksquare E φ : φ exists a path in which ϕ is true;

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters... Motivation and

Intuition How to

Syntax of CTL Semantics of CTL

CTL Some examples or

what we can say

More about

More about semantics Equivalences

The propositional operators: \neg , \lor , \land , \rightarrow have the same meaning of in the propositional logic.

The temporal operators can be read (if φ is a formula) as follows:

- A φ : φ is true in all possible paths;
- \blacksquare E φ : φ exists a path in which ϕ is true;

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and

How to

Syntax of CTL Semantics of CTL

Some examples of what we can say

More about semantics Equivalences

The propositional operators: \neg , \lor , \land , \rightarrow have the same meaning of in the propositional logic.

The temporal operators can be read (if φ is a formula) as follows:

- A φ : φ is true in all possible paths;
- \blacksquare E φ : φ exists a path in which ϕ is true;

The path-specific operators can be read, considering φ and ψ formulas, as:

■ $X\varphi$: φ is true until next state;

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and

Intuition How to

Syntax of CTL Semantics of CTL

CTL Some examples of what we can say

More about semantics Equivalences The propositional operators: \neg , \lor , \land , \rightarrow have the same meaning of in the propositional logic.

The temporal operators can be read (if φ is a formula) as follows:

- A φ : φ is true in all possible paths;
- \blacksquare E φ : φ exists a path in which ϕ is true;

- $\mathbf{X}\varphi : \varphi$ is true until next state;
- $F\varphi$: There is some state in the future where φ is true;

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and

How to communicate Syntax of CTL

Semantics of CTL

Some examples o what we can say More about

More about semantics Equivalences

The propositional operators: \neg , \lor , \land , \rightarrow have the same meaning of in the propositional logic.

The temporal operators can be read (if φ is a formula) as follows:

- $A\varphi$: φ is true in all possible paths;
- \blacksquare E φ : φ exists a path in which ϕ is true;

- $X\varphi : \varphi$ is true until next state;
- $F\varphi$: There is some state in the future where φ is true;
- $G\varphi$: Globally (in all future states) φ is true;

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and

How to communicate

Syntax of CTL Semantics of CTL

Some examples of what we can say

More about semantics Equivalences The propositional operators: \neg , \lor , \land , \rightarrow have the same meaning of in the propositional logic.

The temporal operators can be read (if φ is a formula) as follows:

- A φ : φ is true in all possible paths;
- \blacksquare E φ : φ exists a path in which ϕ is true;

- $\mathbf{X}\varphi : \varphi$ is true until next state;
- $\mathbf{F}\varphi$: There is some state in the future where φ is true;
- $G\varphi$: Globally (in all future states) φ is true;
- $\blacksquare \varphi U \psi$: φ is true at least until ψ becomes true;

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and

Intuition How to

Syntax of CTL

Semantics of CTL

What we can say
More about
semantics

■ Notice that, in CTL, the combination of path specific operators and temporal operators are atomic, i.e., AF is a operator that can be read as "In all paths in the future there is some state where..."

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and

How to

Syntax of CTL Semantics of

CTL Some examples of what we can say

More about semantics Equivalences

- Notice that, in CTL, the combination of path specific operators and temporal operators are atomic, i.e., AF is a operator that can be read as "In all paths in the future there is some state where..."
- Notice as well that the binary operators $A[\varphi U\psi]$ and $E[\varphi U\psi]$ can be represented as AU

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and

How to communicate

Syntax of CTL Semantics of CTL

what we can say

More about
semantics

■ Notice that, in CTL, the combination of path specific operators and temporal operators are atomic, i.e., AF is a operator that can be read as "In all paths in the future there is some state where..."

- Notice as well that the binary operators $A[\varphi U\psi]$ and $E[\varphi U\psi]$ can be represented as AU
- We assume that, similarly to the ¬ operator, the "new" unary operators (AX, EX, AF, EF, AG, and EG) have the first precedence. Next comes the ∧ and ∨ operators. And at last the →, AU and EU

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and

How to communicate

Syntax of CTL Semantics of CTL

what we can say

More about
semantics

■ Notice that, in CTL, the combination of path specific operators and temporal operators are atomic, i.e., AF is a operator that can be read as "In all paths in the future there is some state where..."

- Notice as well that the binary operators $A[\varphi U\psi]$ and $E[\varphi U\psi]$ can be represented as AU
- We assume that, similarly to the ¬ operator, the "new" unary operators (AX, EX, AF, EF, AG, and EG) have the first precedence. Next comes the ∧ and ∨ operators. And at last the →, AU and EU

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Syntax of CTL

Semantics of CTL

- Examples of well-formed formulas:
 - \blacksquare AG($p \lor EFq$)

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...
Motivation and

Motivation and Intuition

How to communicate

Syntax of CTL Semantics of CTL

Some examples of what we can say

More about semantics Equivalences

- Examples of well-formed formulas:
 - $AG(p \lor EFq)$
 - $\blacksquare \ \mathtt{AX}(q \to \mathtt{E}[(p \lor q)\mathtt{U}r])$

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...

Motivation a

How to

Syntax of CTL Semantics of CTL

what we can say

More about
semantics

- Examples of well-formed formulas:
 - $AG(p \lor EFq)$
 - $\blacksquare \ \mathtt{AX}(q \to \mathtt{E}[(p \lor q)\mathtt{U}r])$
 - EFEGp o AFr Note that this is binded as (EFEGp) o AFr, not as EFEG(p o AFr)

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Syntax of CTL Semantics of CTL

what we can sav

- Examples of well-formed formulas:
 - \blacksquare AG($p \lor EFq$)
 - \blacksquare AX $(q \to E[(p \lor q)Ur])$
 - $\mathsf{EFEG}p o \mathsf{AF}r$ Note that this is binded as $(\mathsf{EFEG}p) o \mathsf{AF}r$, not as EFEG($p \rightarrow AFr$)
- Example of formulas that are not well-formed:
 - \blacksquare A \neg G \neg p

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...

Motivation ar

How to

Syntax of CTL Semantics of CTL

Some examples of what we can say More about

More about semantics Equivalences

- Examples of well-formed formulas:
 - $\blacksquare \ \operatorname{AG}(p \vee \operatorname{EF}q)$
 - $\blacksquare \ \mathtt{AX}(q \to \mathtt{E}[(p \lor q)\mathtt{U}r])$
 - EFEG $p o ext{AF}r$ Note that this is binded as (EFEGp) $o ext{AF}r$, not as EFEG $(p o ext{AF}r)$
- Example of formulas that are not well-formed:
 - A \neg G \neg p
 - \blacksquare F[pUs]

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...

Motivation ar

Intuition

Syntax of CTL

Semantics of CTL Some examples of

What we can say
More about
semantics
Equivalences

- Examples of well-formed formulas:
 - $AG(p \lor EFq)$
 - $\blacksquare \ \mathtt{AX}(q \to \mathtt{E}[(p \lor q)\mathtt{U}r])$
 - EFEGp o AFr Note that this is binded as (EFEGp) o AFr, not as EFEG(p o AFr)
- Example of formulas that are not well-formed:
 - A¬G¬p
 - $\blacksquare F[pUs]$
 - $A[pUs \land qUs]$

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Semantics of CTL

what we can sav

Different from usual logics, CTL formulas are interpreted by a transition system. Given an set of atoms:

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters... Motivation an Intuition

How to communicate Syntax of CT Semantics of CTL

Some examples of what we can say More about semantics

Different from usual logics, CTL formulas are interpreted by a transition system. Given an set of atoms:

Definition (1)

A transition system \mathcal{M} is a triple $\mathcal{M}=(S,\to,L)$ in which S is a set of states, \to is a binary relation over S ($\to\subseteq S\times S$) and $L:S\to \mathcal{P}(Atoms)$ is a labelling function.

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters... Motivation and Intuition How to

How to communicate Syntax of CTI Semantics of CTL

Some examples o what we can say More about semantics Equivalences Different from usual logics, CTL formulas are interpreted by a transition system. Given an set of atoms:

Definition (1)

A transition system \mathcal{M} is a triple $\mathcal{M}=(S,\to,L)$ in which S is a set of states, \to is a binary relation over S ($\to\subseteq S\times S$) and $L:S\to\mathcal{P}(Atoms)$ is a labelling function.

Definition (2)

A **model** is a duple \mathcal{M}, s in which \mathcal{M} is a transition system and s is a state.

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters... Motivation and Intuition How to

Intuition
How to
communicate
Syntax of CT
Semantics of
CTL

Some examples of what we can say More about semantics Equivalences

Different from usual logics, CTL formulas are interpreted by a transition system. Given an set of atoms:

Definition (1)

A transition system \mathcal{M} is a triple $\mathcal{M}=(S,\to,L)$ in which S is a set of states, \to is a binary relation over S ($\to\subseteq S\times S$) and $L:S\to\mathcal{P}(Atoms)$ is a labelling function.

Definition (2)

A **model** is a duple \mathcal{M}, s in which \mathcal{M} is a transition system and s is a state.

Notation: we will use $\mathcal{M}, s \vDash \varphi$ to denote that the model \mathcal{M}, s satisfies the formula φ

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and Intuition How to communicate Syntax of CTL Semantics of CTL

what we can say

More about
semantics
Equivalences

Take an arbitrary model \mathcal{M} . Let s, s_1, s_2, s_3 be states in S. Let $\varphi, \varphi_1, \varphi_2$ be well-formed formulas of CTL. And let p be an atom. The satisfaction of CTL formulas can be defined as follows:

■ \mathcal{M} , $s \vDash \top$ and \mathcal{M} , $s \not\vDash \bot$ for all $s \in S$

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and Intuition How to communicate Syntax of CTL Semantics of

Some examples of what we can say More about semantics
Equivalences

Take an arbitrary model \mathcal{M} . Let s, s_1, s_2, s_3 be states in S. Let $\varphi, \varphi_1, \varphi_2$ be well-formed formulas of CTL. And let p be an atom. The satisfaction of CTL formulas can be defined as follows:

- $\mathcal{M}, s \vDash \top$ and $\mathcal{M}, s \not\vDash \bot$ for all $s \in S$
- \mathcal{M} , $s \models p$ iff $p \in L(S)$

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and Intuition
How to communicate Syntax of CTL Semantics of

Some examples of what we can say More about semantics

Equivalences

Take an arbitrary model \mathcal{M} . Let s, s_1, s_2, s_3 be states in S. Let $\varphi, \varphi_1, \varphi_2$ be well-formed formulas of CTL. And let p be an atom. The satisfaction of CTL formulas can be defined as follows:

- $\mathcal{M}, s \vDash \top$ and $\mathcal{M}, s \not\vDash \bot$ for all $s \in S$
- $\blacksquare \mathcal{M}, s \vDash p \text{ iff } p \in L(S)$
- $\blacksquare \mathcal{M}, s \vDash \neg \varphi \text{ iff } \mathcal{M}, s \not\vDash \varphi$

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and Intuition How to communicate Syntax of CTL Semantics of

Some examples o what we can say More about semantics Take an arbitrary model \mathcal{M} . Let s, s_1, s_2, s_3 be states in S. Let $\varphi, \varphi_1, \varphi_2$ be well-formed formulas of CTL. And let p be an atom. The satisfaction of CTL formulas can be defined as follows:

- $\mathcal{M}, s \vDash \top$ and $\mathcal{M}, s \not\vDash \bot$ for all $s \in S$
- \mathcal{M} , $s \models p$ iff $p \in L(S)$
- $\blacksquare \mathcal{M}, s \vDash \neg \varphi \text{ iff } \mathcal{M}, s \not\vDash \varphi$
- $\mathcal{M}, s \vDash \varphi_1 \land \varphi_2$ iff $\mathcal{M}, s \vDash \varphi_1$ AND $\mathcal{M}, s \vDash \varphi_2$

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and Intuition How to communicate Syntax of CTL Semantics of

Some examples of what we can say More about semantics

- \blacksquare $\mathcal{M}, s \vDash \top$ and $\mathcal{M}, s \not\vDash \bot$ for all $s \in S$
- $\blacksquare \mathcal{M}, s \vDash p \text{ iff } p \in L(S)$
- $\blacksquare \mathcal{M}, s \vDash \neg \varphi \text{ iff } \mathcal{M}, s \not\vDash \varphi$
- $\mathcal{M}, s \vDash \varphi_1 \land \varphi_2$ iff $\mathcal{M}, s \vDash \varphi_1$ AND $\mathcal{M}, s \vDash \varphi_2$
- \mathcal{M} , $s \vDash \varphi_1 \lor \varphi_2$ iff \mathcal{M} , $s \vDash \varphi_1$ OR \mathcal{M} , $s \vDash \varphi_2$

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and Intuition How to communicate Syntax of CTL Semantics of

Some examples of what we can say More about semantics

- \blacksquare $\mathcal{M}, s \vDash \top$ and $\mathcal{M}, s \not\vDash \bot$ for all $s \in S$
- \mathcal{M} , $s \models p$ iff $p \in L(S)$
- \mathcal{M} , $s \vDash \neg \varphi$ iff \mathcal{M} , $s \nvDash \varphi$
- $\blacksquare \mathcal{M}, s \vDash \varphi_1 \land \varphi_2 \text{ iff } \mathcal{M}, s \vDash \varphi_1 \text{ AND } \mathcal{M}, s \vDash \varphi_2$
- \mathcal{M} , $s \vDash \varphi_1 \lor \varphi_2$ iff \mathcal{M} , $s \vDash \varphi_1$ OR \mathcal{M} , $s \vDash \varphi_2$
- $\blacksquare \mathcal{M}, s \vDash \varphi_1 \rightarrow \varphi_2 \text{ iff } \mathcal{M}, s \not\vDash \varphi_1 \text{ OR } \mathcal{M}, s \vDash \varphi_2$

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...
Motivation and Intuition
How to communicate
Syntax of CTI
Semantics of

Some examples of what we can say More about semantics

Take an arbitrary model \mathcal{M} . Let s, s_1, s_2, s_3 be states in S. Let $\varphi, \varphi_1, \varphi_2$ be well-formed formulas of CTL. And let p be an atom. The satisfaction of CTL formulas can be defined as follows:

■ $\mathcal{M}, s \vDash \mathsf{AX}\varphi$ iff for all s_1 that $s \to s_1$ and $\mathcal{M}, s_1 \vDash \varphi$. Thus, AX says: "in every next state..."

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...
Motivation and
Intuition
How to
communicate
Syntax of CTI
Semantics of

Some examples of what we can say More about semantics
Equivalences

- $\mathcal{M}, s \vDash \mathtt{AX}\varphi$ iff for all s_1 that $s \to s_1$ and $\mathcal{M}, s_1 \vDash \varphi$. Thus, AX says: "in every next state..."
- $\mathcal{M}, s \vDash \mathsf{EX}\varphi$ iff exists s_1 that $s \to s_1$ and $M, s_1 \vDash \varphi$. Thus, EX says: "in some next state…"

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Semantics of what we can sav

- $\blacksquare \mathcal{M}, s \vDash \mathtt{AX}\varphi \text{ iff for all } s_1 \text{ that } s \to s_1 \text{ and } \mathcal{M}, s_1 \vDash \varphi.$ Thus, AX says: "in every next state..."
- \mathcal{M} , $s \models \mathsf{EX}\varphi$ iff exists s_1 that $s \to s_1$ and M, $s_1 \models \varphi$. Thus, EX says: "in some next state..."
- $M, s, \models AG\varphi$ iff for all paths $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s=s_1$, for all s_i , $M, s_i \models \varphi$. Thus, AG says: "In all possible paths from now on in all next states..."

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and Intuition
How to communicate
Syntax of CT
Semantics of

Some examples of what we can say More about semantics
Equivalences

- $\mathcal{M}, s \vDash \mathtt{AX}\varphi$ iff for all s_1 that $s \to s_1$ and $\mathcal{M}, s_1 \vDash \varphi$. Thus, AX says: "in every next state..."
- $\mathcal{M}, s \models \mathsf{EX}\varphi$ iff exists s_1 that $s \to s_1$ and $M, s_1 \models \varphi$. Thus, EX says: "in some next state..."
- $M, s, \models AG\varphi$ iff for all paths $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, for all s_i , $M, s_i \models \varphi$. Thus, AG says: "In all possible paths from now on in all next states..."
- $M, s, \models AG\varphi$ iff exists some path $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, for all s_i , $M, s_i \models \varphi$ Thus, EG says: "Exists a path from now on in all next states..."

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and Intuition
How to communicate
Syntax of CT
Semantics of

Some examples of what we can say More about semantics
Equivalences

- $\mathcal{M}, s \vDash \mathtt{AX}\varphi$ iff for all s_1 that $s \to s_1$ and $\mathcal{M}, s_1 \vDash \varphi$. Thus, AX says: "in every next state..."
- $\mathcal{M}, s \models \mathsf{EX}\varphi$ iff exists s_1 that $s \to s_1$ and $M, s_1 \models \varphi$. Thus, EX says: "in some next state..."
- $M, s, \models AG\varphi$ iff for all paths $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, for all s_i , $M, s_i \models \varphi$. Thus, AG says: "In all possible paths from now on in all next states..."
- $M, s, \models AG\varphi$ iff exists some path $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, for all s_i , $M, s_i \models \varphi$ Thus, EG says: "Exists a path from now on in all next states..."

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters... Motivation and Intuition How to

How to communicate Syntax of CTL CTL

What we can say
More about
semantics
Equivalences

Take an arbitrary model \mathcal{M} . Let s, s_1, s_2, s_3 be states in S. Let $\varphi, \varphi_1, \varphi_2$ be well-formed formulas of CTL. And let p be an atom. The satisfaction of CTL formulas can be defined as follows:

■ $M, s, \models AF\varphi$ iff for all paths $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, exists s_i , $M, s_i \models \varphi$. Thus, AF says: "In all possible paths from now on, in some next state..."

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...
Motivation and Intuition
How to

Syntax of CTL
Semantics of
CTL
Some examples

what we can say
More about
semantics
Equivalences

- $M, s, \models AF\varphi$ iff for all paths $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, exists s_i , $M, s_i \models \varphi$. Thus, AF says: "In all possible paths from now on, in some next state..."
- $M, s, \models \text{EF}\varphi$ iff exists some path $s_1 \to s_2 \to s_3 \to ...$ in which $s = s_1$, that exists s_i , $M, s_i \models \varphi$. Thus, EF says: "In some path from now on, in some next state..."

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...
Motivation and Intuition
How to communicate
Syntax of CTI
Semantics of

Some examples of what we can say More about

More about semantics Equivalences

- $M, s, \models AF\varphi$ iff for all paths $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, exists s_i , $M, s_i \models \varphi$. Thus, AF says: "In all possible paths from now on, in some next state..."
- $M, s, \models \text{EF}\varphi$ iff exists some path $s_1 \to s_2 \to s_3 \to ...$ in which $s = s_1$, that exists s_i , $M, s_i \models \varphi$. Thus, EF says: "In some path from now on, in some next state..."
- $M, s, \models A[\varphi_1 U \varphi_2]$ iff for all paths $s_1 \to s_2 \to s_3 \to ...$ in which $s = s_1$, this path satisfies $\varphi_1 U \varphi_2$, i.e., exists s_i in the path such that $M, s_i \models \varphi_2$ and, for all j < i, $M, s_j \models \varphi_1$. Thus, AU says: "For all paths from now on, until some state..."

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation an Intuition How to communicate Syntax of CT Semantics of

what we can say More about semantics Equivalences

- $M, s, \models AF\varphi$ iff for all paths $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, exists s_i , $M, s_i \models \varphi$. Thus, AF says: "In all possible paths from now on, in some next state..."
- $M, s, \models \text{EF}\varphi$ iff exists some path $s_1 \rightarrow s_2 \rightarrow s_3 \rightarrow ...$ in which $s = s_1$, that exists s_i , $M, s_i \models \varphi$. Thus, EF says: "In some path from now on, in some next state..."
- $M, s, \models A[\varphi_1 U \varphi_2]$ iff for all paths $s_1 \to s_2 \to s_3 \to ...$ in which $s = s_1$, this path satisfies $\varphi_1 U \varphi_2$, i.e., exists s_i in the path such that $M, s_i \models \varphi_2$ and, for all j < i, $M, s_j \models \varphi_1$. Thus, AU says: "For all paths from now on, until some state..."
- $M, s, \models \mathtt{E}[\varphi_1 \mathtt{U} \varphi_2]$ iff exists some path $s_1 \mapsto s_2 \mapsto s_3 \mapsto \ldots$

Examples

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

chapters...
Motivation and
Intuition
How to
communicate
Syntax of CTI
Semantics of

Some examples of what we can say

More about semantics
Equivalences

■ "It's possible to get to a state where something has started but it's not ready": $EF(started \land \neg ready)$

Examples

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and Intuition How to communicate Syntax of CTI Semantics of CTL

Some examples of what we can say

More about semantics Equivalences

- "It's possible to get to a state where something has started but it's not ready": EF(started ∧ ¬ready)
- "A certain process is enabled infinitely often on every computation path": AG(AFenabled)

Examples

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

Motivation and Intuition How to communicate Syntax of CTL Semantics of

Some examples of what we can say

More about semantics
Equivalences

- "It's possible to get to a state where something has started but it's not ready": EF(started ∧ ¬ready)
- "A certain process is enabled infinitely often on every computation path": AG(AFenabled)
- "An upwards travelling lift at the second floor does not change its direction when it has passengers wishing to go to the fifth floor":

 $\texttt{AG}(\textit{floor2} \land \textit{directionUp} \land \textit{button5} \rightarrow \texttt{A}[\textit{directionUp} \texttt{Ufloor5}])$

Equivalences

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters... Motivation ar Intuition

How to communicate Syntax of CT Semantics of CTL

Some examples o what we can say

More about semantics Equivalences

Definition

Two CTL formulas φ and ψ are said to be **semantically equivalent** if any state in any model which satisfies one of them also satisfies the other;

Equivalences

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters... Motivation an Intuition

How to communicate Syntax of CT Semantics of CTL

What we can say
More about
semantics
Equivalences

Definition

Two CTL formulas φ and ψ are said to be **semantically equivalent** if any state in any model which satisfies one of them also satisfies the other;

Notation: we denote the equivalence of φ and ψ by $\varphi \equiv \psi$

Example of equivalences

Computation Tree Logic

Luis Tertulino & Ronaldo Silveira

In previous chapters...

Motivation an

How to communicate Syntax of CT Semantics of CTL

Some examples of what we can say

More about semantics

Let φ be an arbitrary CTL formula.

$$\quad \blacksquare \ \neg \mathtt{AF} \varphi \equiv \mathtt{EG} \neg \varphi$$

$$\quad \blacksquare \ \neg \mathrm{EF} \varphi \equiv \mathrm{AG} \neg \varphi$$

$$\quad \blacksquare \ \neg \mathtt{AX} \varphi \equiv \mathtt{EX} \neg \varphi$$

$$\blacksquare \ \mathsf{AF}\varphi \equiv \mathsf{A}[\top \mathsf{U}\varphi]$$

$$\blacksquare \ \mathrm{EF}\varphi \equiv \mathrm{E}[\top \mathrm{U}\varphi]$$