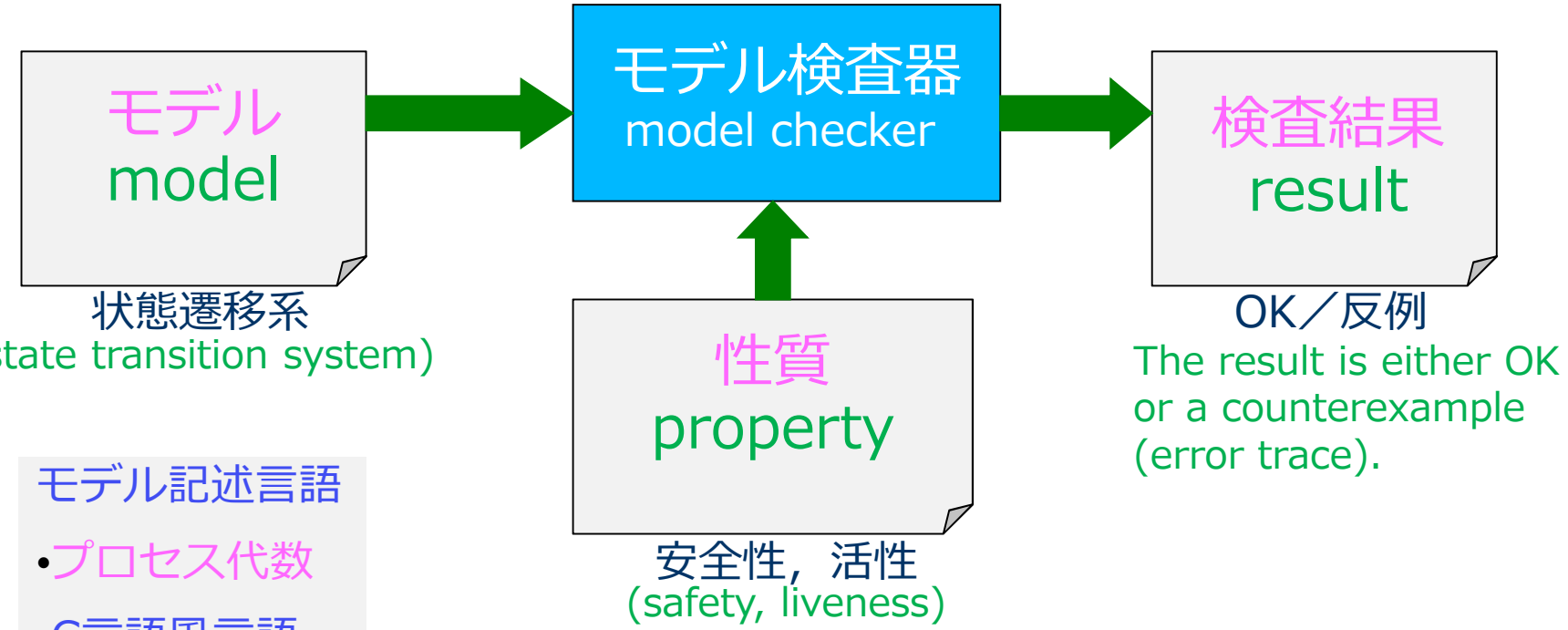
知能ソフトウェア特論
Intelligent Software

モデル検査（２）

時相論理
Model Checking (2)
Temporal Logic

モデル検査器の概要 (Overview of model checker)



モデル記述言語

- プロセス代数
- C言語風言語

Model description language

- Process algebra
- C-like language

性質記述言語

- 時相論理

Property description language

- Temporal logic

１．時相論理の概要 (Overview of temporal logic)

逐次プログラムの検証
古典論理による入出力関係の表現で十分

To verify sequential programs, it is enough to represent the systems by input/output relationships in classical logic.

並行システムの検証
通常は非停止．内部の状態遷移の表現が重要．

To verify concurrent systems, which are often non-terminating, it is important to represent the internal state transitions of the systems in non-classical logic.

時相論理の特徴 (Characteristics of temporal logic)

時相論理

Temporal logic allows you to

- 並行システムの性質を記述できる
  describe properties of concurrent systems,
- 時間にかかわる表現が可能
  describe properties related to time,
- 時相演算子で状態遷移の性質を表現
  describe properties of state transition using temporal operators.
- 線形時相論理 LTL
  LTL: Linear Temporal Logic
- 計算木時相論理CTL
  CTL: Computational Tree Logic

2. 状態遷移系 (State transition systems)

$V = \{v_1, \ldots, v_n\}$ ： 並行システムにおけるシステム変数の集合
(the set of variables in a concurrent system)

D ： 各変数がとる値の有限集合（領域）
(domain: a finite set of values taken by the variables)

$s = (a_1, \ldots, a_n)$ ： システム変数のとる値（状態）のベクトル
(state: a vector of values taken by the variables)

AP： 状態から真偽値が定まる原始命題の集合

例:「$v_1 = 3$」は原始命題

(the set of atomic propositions: Their truth values are determined in each state. For example, "v_1=3" is an atomic proposition whose value is true in the states where the value of v_1 is 3.)

状態遷移系の数学的定義
(Mathematical definition of state transition systems)

状態遷移系 $M = (S, S_0, R, L)$

State transition systems are formally defined as a 4-tuple.

S：状態の有限集合 (S is a finite set of states)

S_0：初期状態の集合 ($S_0 \subseteq S$ is a set of initial states)

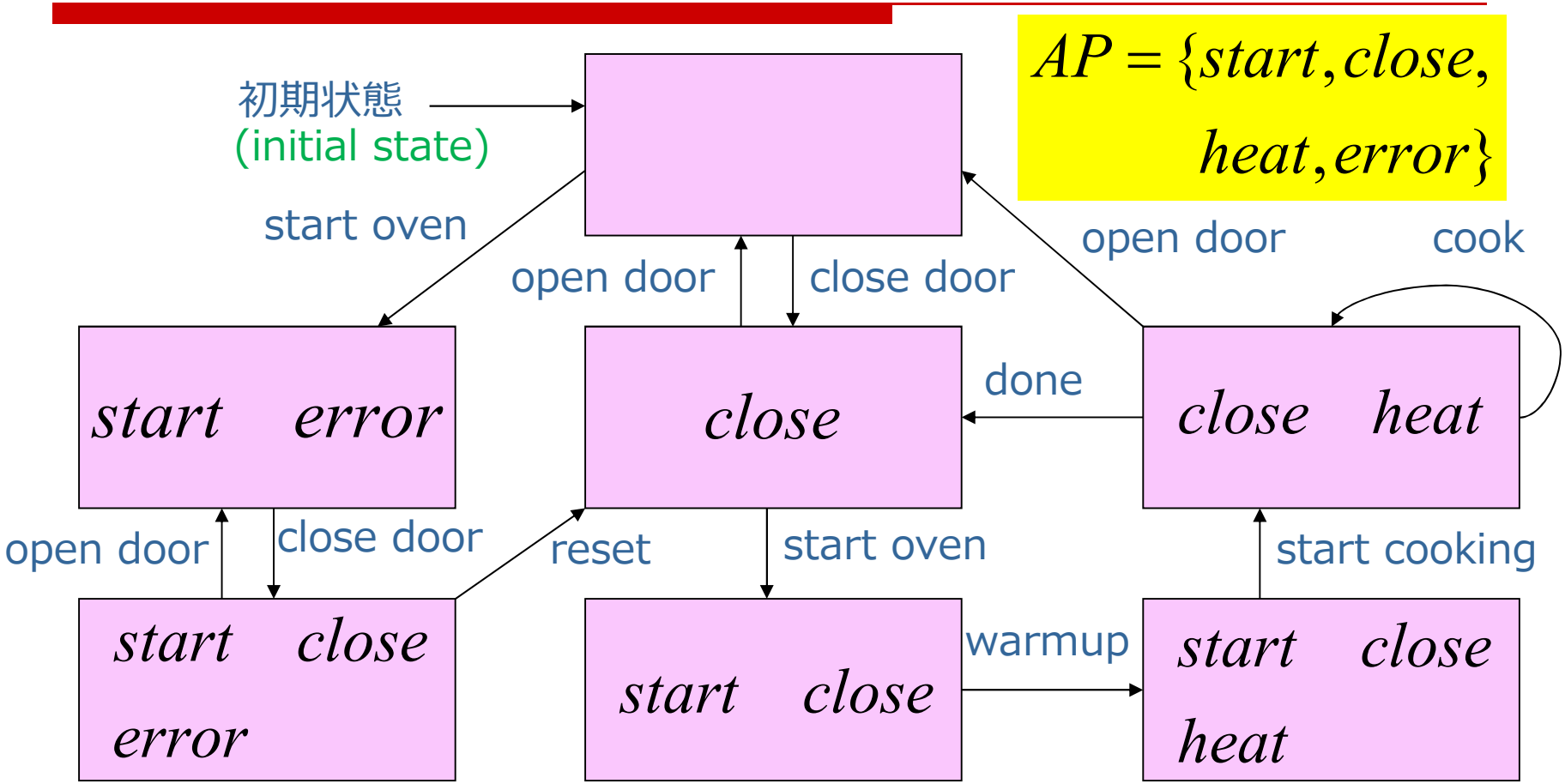
R：状態遷移関係 ($R \subseteq S \times S$ is a state transition relation)

L：ラベリング関数
（各状態に「その状態で真である原始命題の集合」
をラベルとして付けるための関数）

(L: $S \rightarrow 2^{AP}$ is a labeling function, which assigns each state in S with a label, i.e. a subset of AP that are true in that state, where 2^{AP} is a powerset of AP, i.e., the set of all subsets of AP)

状態遷移系の例（オーブンレンジ）

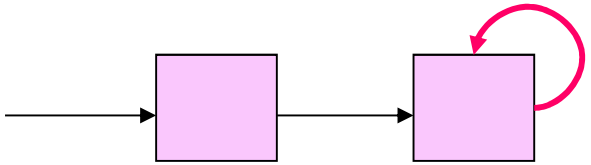
(An example of a state transition system: an oven)



パス (Path)

どの状態 s においても，必ず遷移先の状態 s′ があるものとする．
この条件が成り立たないときは，s から s′ 自身に遷移可能なように
R を修正．

We assume every state s has a state s' to which transition from s is possible. Otherwise, we will fix R so that transition from s to s itself is possible.

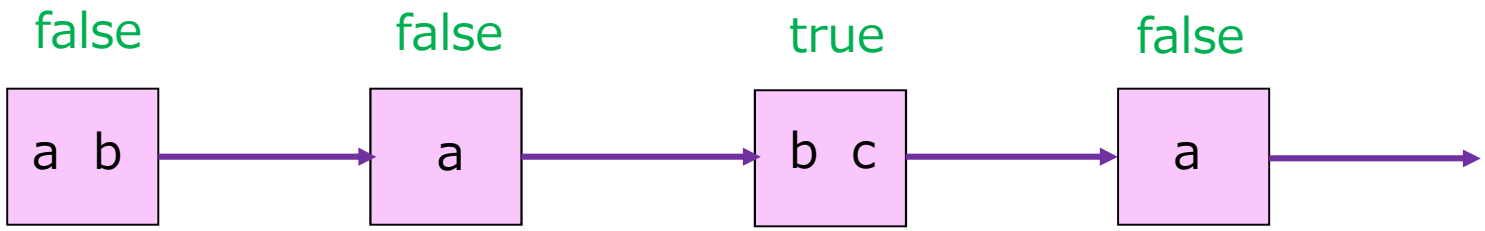


状態 s からのパスとは，状態 s から始まる状態の無限列
$\pi = (s\ s_1\ s_2\ \cdots)$ のこと

A path from a state s is an infinite sequence of states $\pi = (s_0 s_1 s_2 \cdots)$, where $s_0 = s$ and $(s_i, s_{i+1}) \in R$ for all i.

3．線形時相論理 LTL (Linear Temporal Logic: LTL)

Is c true in this path?



このパスでは，いつか必ず c が成り立つ

LTLでは，この性質を **F**c で表す

In this path, **c** will be eventually true.
In LTL, this property is expressed as **F**c.

LTLの構文論 (Syntax of LTL)

LTL式 (LTL formulas)

- p が原始命題ならば，p はLTL式

  An atomic proposition *p* is an LTL formula.

- f, g がLTL式ならば

  $$\neg f, f \vee g, f \wedge g, f \rightarrow g$$

  もそれぞれLTL式

- f, g がLTL式ならば

  $$\mathbf{X}f, \mathbf{F}f, \mathbf{G}f, f\,\mathbf{U}g$$

  もそれぞれLTL式

These eight formulas are LTL formulas, if *f* and *g* are LTL formulas.

LTL式の構文を時相演算子 **X**,**F**,**G**,**U** を用いて帰納的に定義
The syntax for LTL formulas are defined inductively by using temporal operators **X**, **F**, **G**, and **U**.

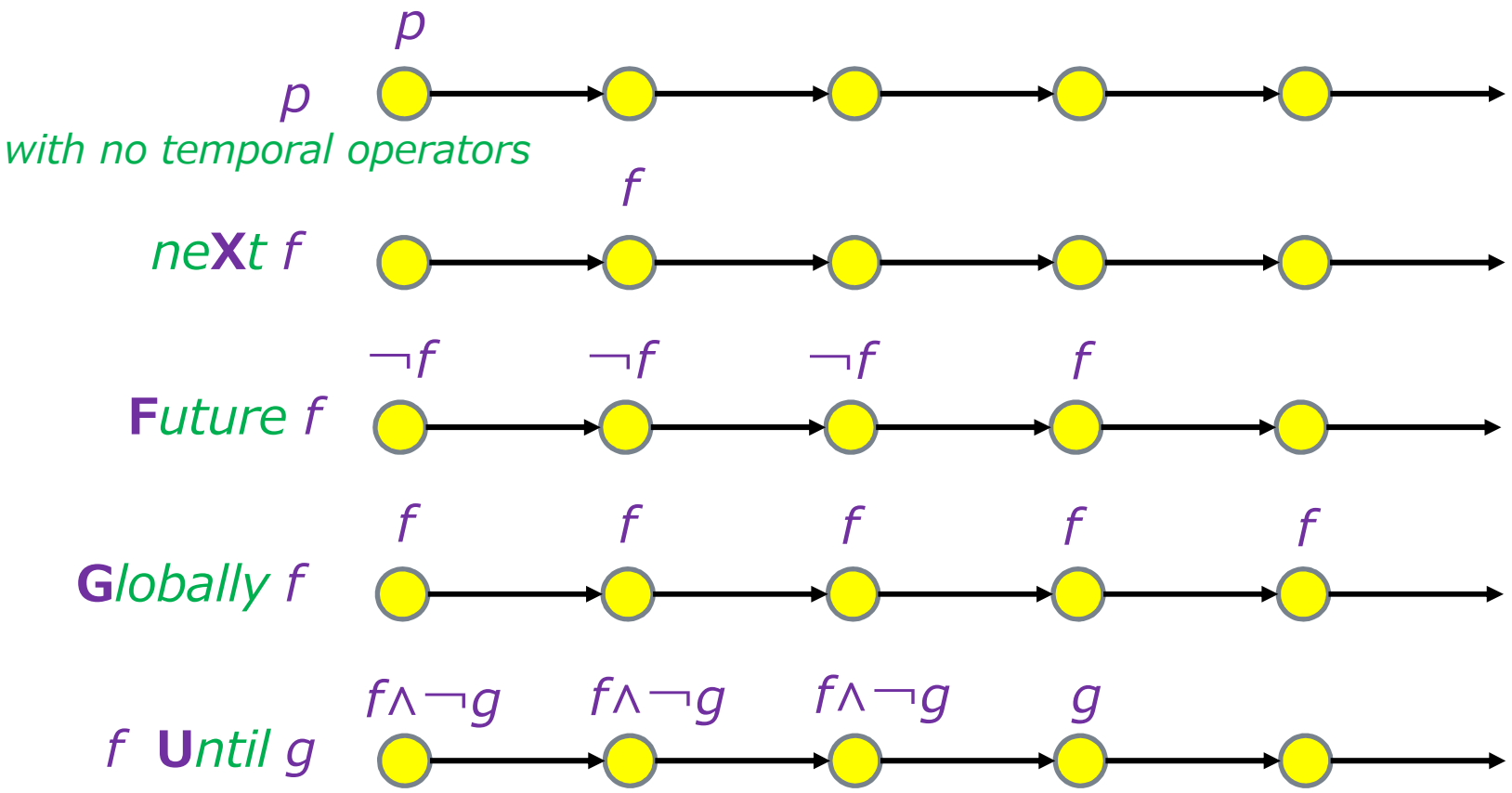
LTLの直観的な意味(1/2) (Intuitive meaning of LTL)

LTL式の真偽は，状態遷移系上のパスごとに定まる.

The truth of an LTL formula is determined for each path.

$\mathbf{X}f$	neXt time	パス上の次の状態で f が成り立つ f is true in the *neXt* state on the path.
$\mathbf{F}f$	Finally, Future, eventually	パス上のある状態でいつかは f が成り立つ f will be *F*inally true in some state on the path.
$\mathbf{G}f$	Globally, always	パス上のすべての状態でいつも f が成り立つ f is true in all the states (*Globally*) on the path.
$f\,\mathbf{U}g$	Until	パス上のある状態で g が成り立ち，かつ その直前までのすべての状態で f が成り立つ g is true in some state s on the path, and f is true in all the states before s (*Until* g is true).

LTLの直観的な意味(2/2)



LTLモデル検査器　(LTL Model Checker)

SPIN など

LTLモデル検査器

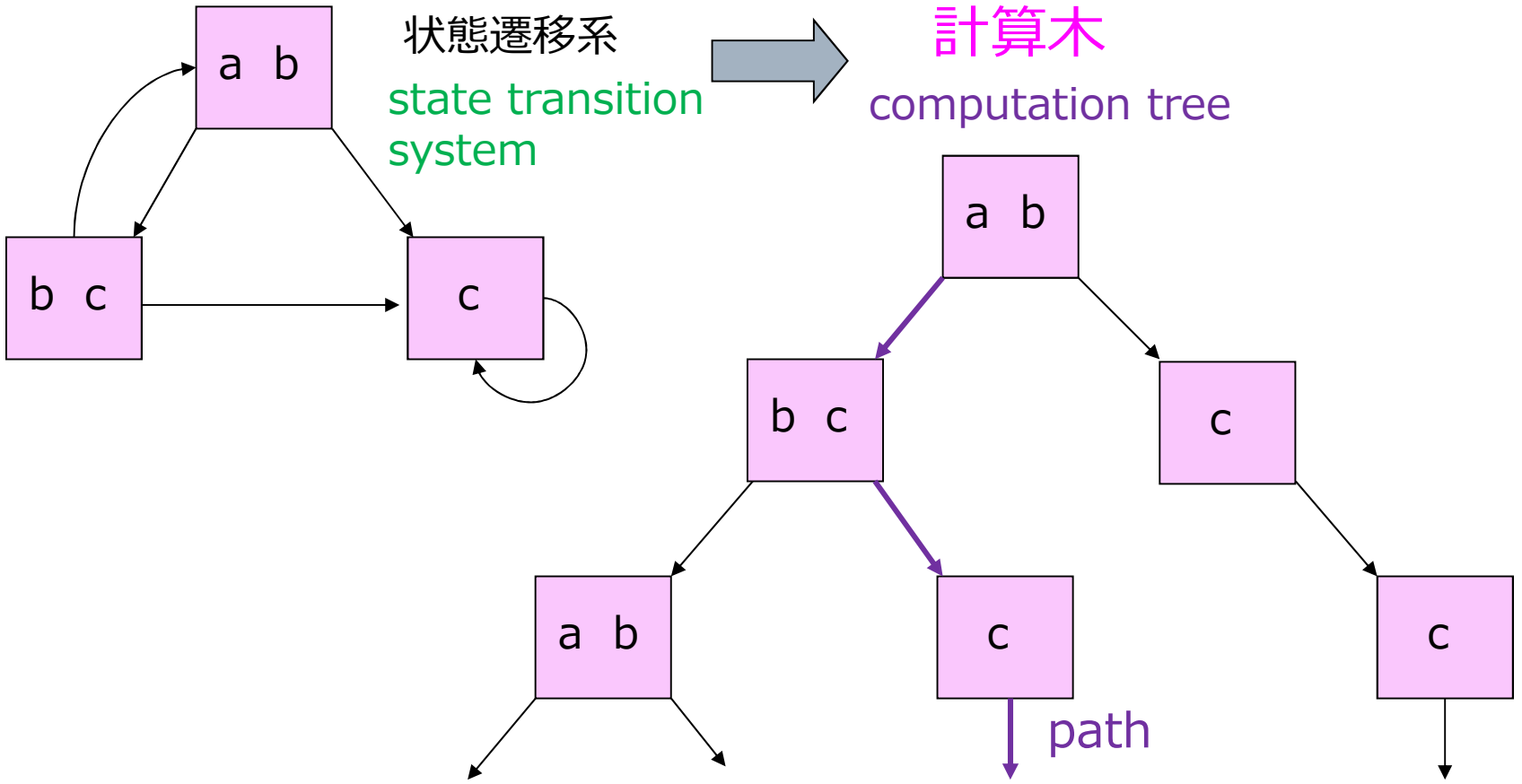
　モデルおよびLTL式 *f* が入力されると，そのモデルの初期状態から始まるすべてのパスについて，性質 *f* が成り立つかどうかを検査する．

　f が成り立たないようなパスがあれば，それを反例として出力する．

When an LTL formula *f* is input to an LTL model checker such as SPIN, it checks all the paths p starting from the initial states to see if *f* is true on *p*.

If it finds a path on which *f* is not true, it will output that path.

４． 計算木時相論理 CTL (Computation Tree Logic: CTL)



CTLの構文論　(Syntax of CTL)

CTLでは，LTLの時相演算子に加えて，パス限量子 EとAを用いる

(CTL uses the path quantifiers **E** and **A** in addition to LTL temporal operators)

CTL式 (CTL formulas)

- p が原始命題ならば，p はCTL式

  An atomic proposition *p* is a CTL formula.

- f, g がCTL式ならば

  $\neg f, f \vee g, f \wedge g, f \rightarrow g$

  もそれぞれCTL式

  These 12 formulas are CTL formulas, if *f* and *g* are CTL formulas.

- f, g がCTL式ならば

  $\mathbf{EX}f, \mathbf{EF}f, \mathbf{EG}f, \mathbf{E}(f\,\mathbf{U}g), \quad \mathbf{AX}f, \mathbf{AF}f, \mathbf{AG}f, \mathbf{A}(f\,\mathbf{U}g)$

  もそれぞれCTL式

CTLの直観的な意味： E,A

(Intuitive meaning of CTL: E,A)

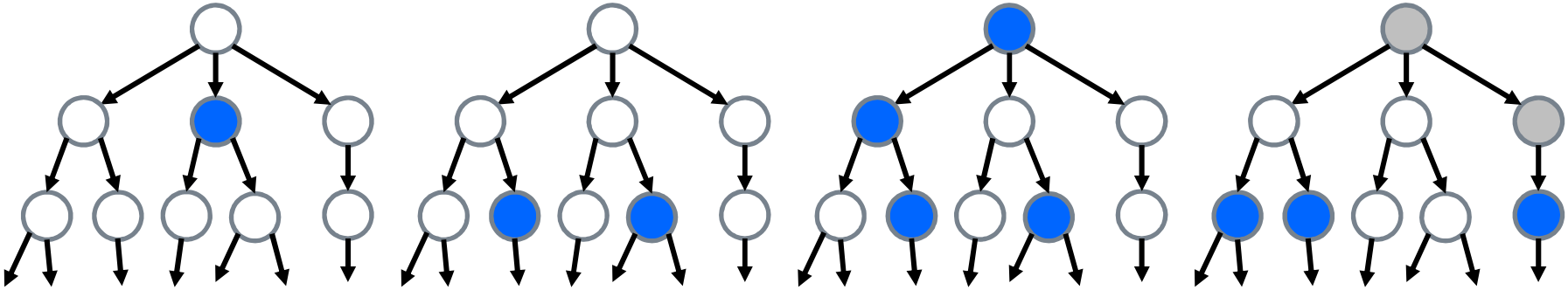
CTL式の真偽は，着目する状態から始まる計算木ごとに定まる

The truth of a CTL formula is determined for each computation tree starting from a state.

E*f*	there **E**xists a path	この状態から始まる**ある**パスにおいて *f* が成り立つ. There exists a path in the computation tree for which *f* is true.
A*f*	for **A**ll paths	この状態から始まる**すべての**パスにおいて*f* が成り立つ *f* is true for all paths in the computation tree.

CTLの直観的な意味： Eの使用例

(Intuitive meaning of CTL: Sample use of E)



EX *blue*

There exists a path such that the next state is blue.

EF *blue*

There exists a path that will, in future, get blue.

EG *blue*

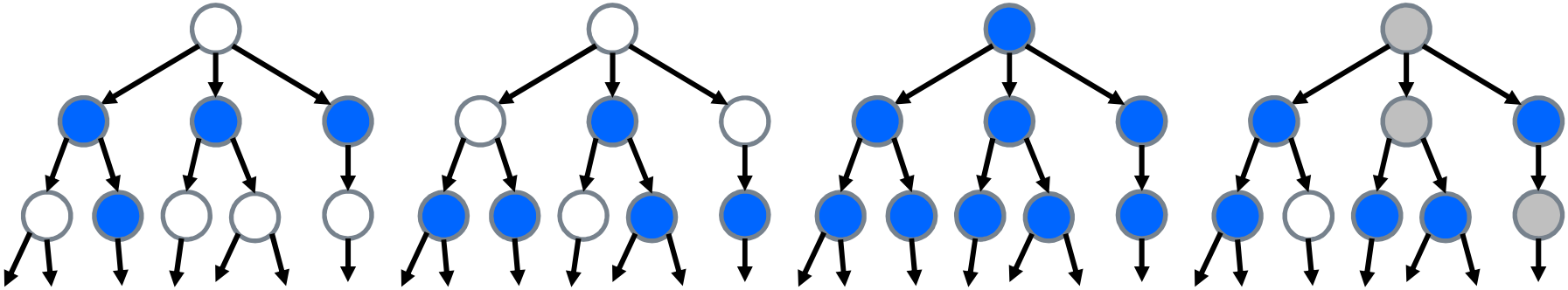
There exists a path that is globally blue.

E(gray **U** *blue*)

There exists a path that is gray until blue.

CTLの直観的な意味: Aの使用例

(Intuitive meaning of CTL: Sample use of A)



AX *blue*

For all paths, the next state is blue.

AF *blue*

For all paths, it will finally become blue.

AG *blue*

For all paths, it is globally blue.

A(gray **U** *blue*)

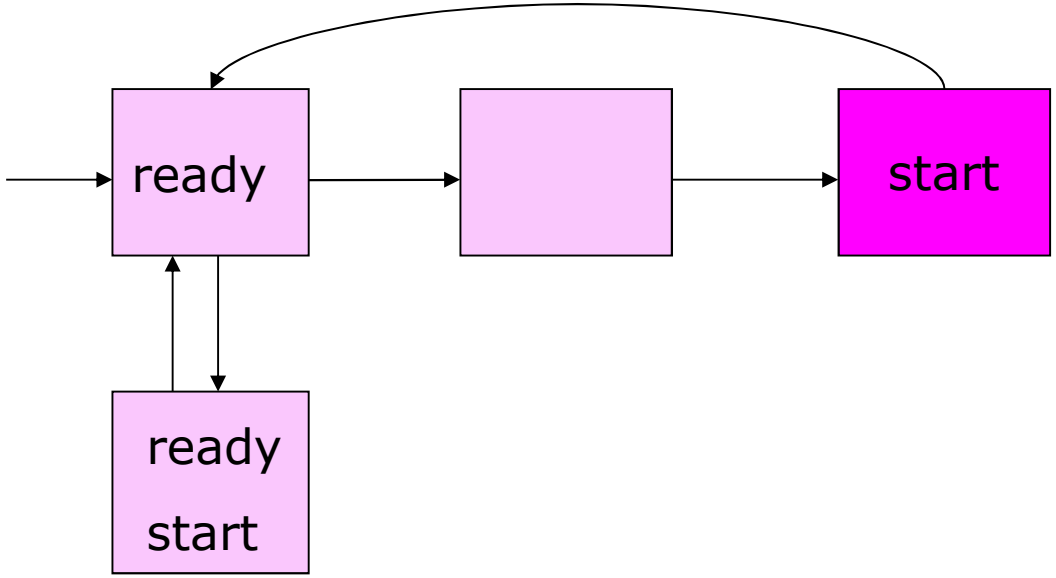
For all paths, it is gray until blue.

CTL式の例(1/4) (Example of CTL formula)

EF($start \wedge \neg ready$)

start は真だが *ready* が偽である状態に至る経路が存在する.

There exists a path where in future we will see that *start* is true but *ready* is not.

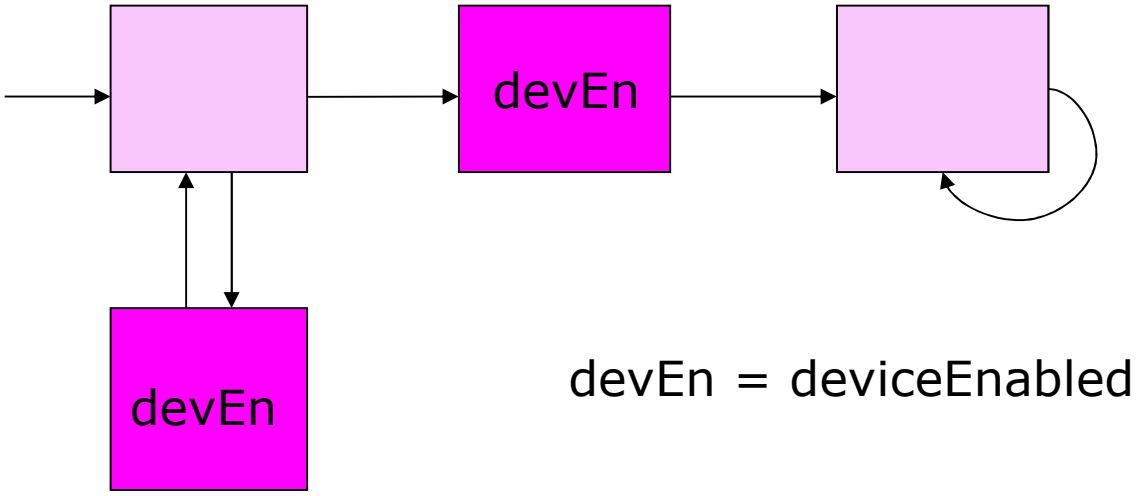


CTL式の例(2/4) (Example of CTL formula)

AF *deviceEnabled*

どの経路においても，いつか *deviceEnabled* が真となる．

For all paths, we will see that *deviceEnabled* will be finally true.

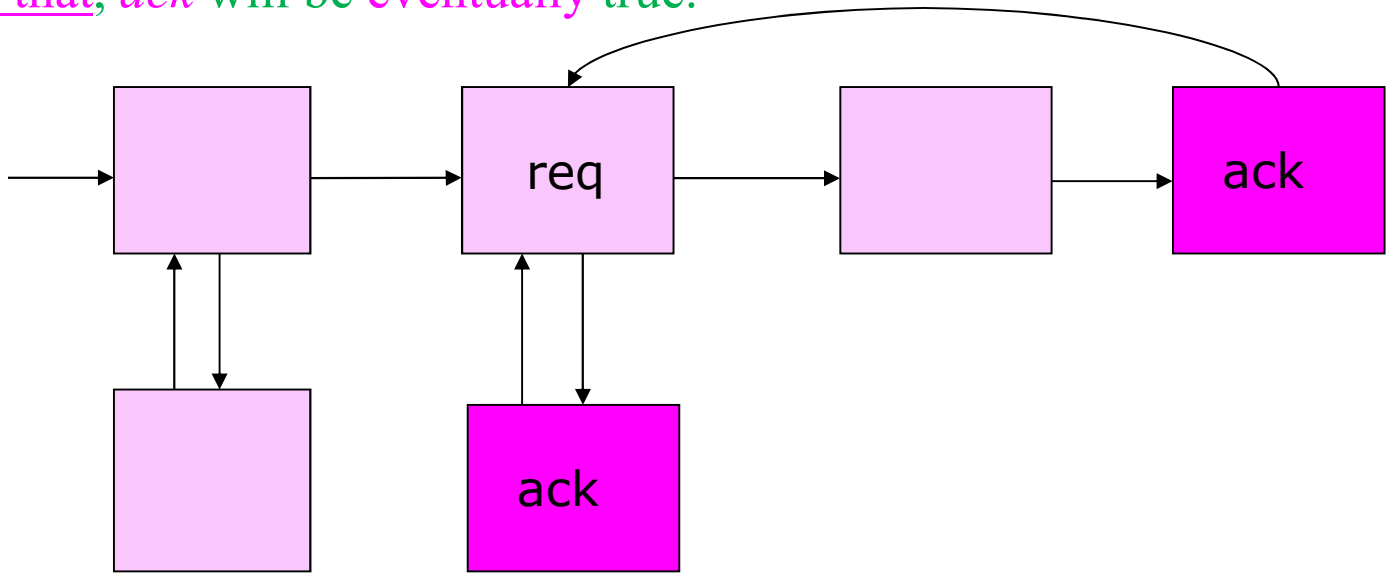


CTL式の例(3/4) (Example of CTL formula)

AG(*req* → **AF** *ack*)

どの経路のどの状態においても，*req* が真ならば，いつかは *ack* が真となる．

In any path, it is always true that if *req* is true at that time, then for all paths after that, *ack* will be eventually true.

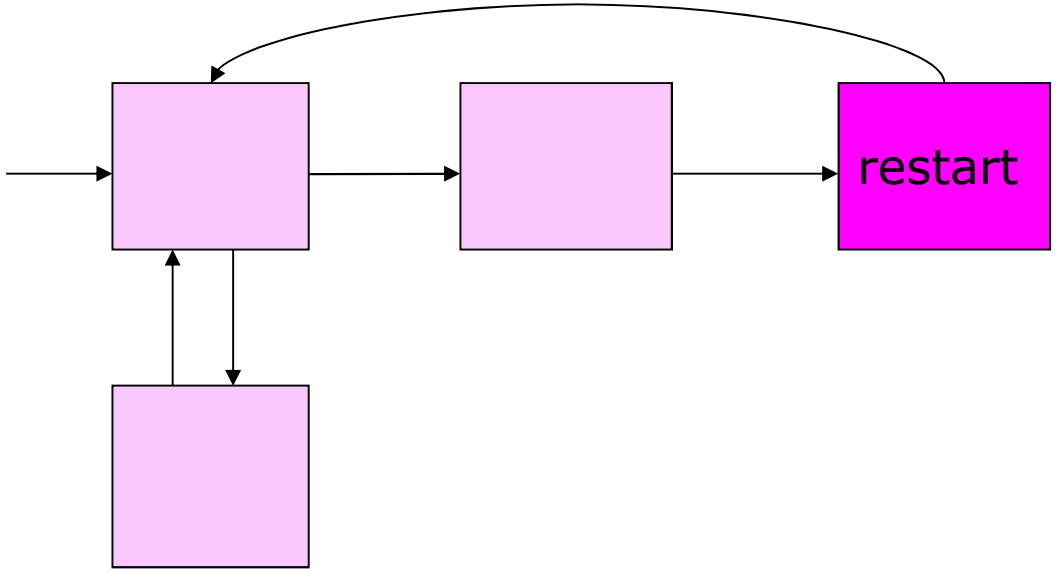


CTL式の例(4/4) (Example of CTL formula)

AG(EF *restart*)

どの状態からも，(経路を適切に選べば) *restart* 状態に到達可能である.

From any state, there is a path to reach a restart state.



演習問題4
Exercise 4

つぎの等式を証明せよ．

(1) $\mathbf{F}f = true\,\mathbf{U}\,f$

(2) $\neg\mathbf{G}f = \mathbf{F}\neg f$

(3) $\neg\mathbf{A}f = \mathbf{E}\neg f$

(4) $\neg\mathbf{X}f = \mathbf{X}\neg f$

Prove the following identical equations.

参考

$$\neg(\forall x)P = (\exists x)\neg P$$

$$\neg(\exists x)P = (\forall x)\neg P$$

参考

以下のスライドは，参考までに，LTLとCTLの形式的な意味論についてやや詳しく書いたものです

（授業では使いません）

LTLの形式的意味論(1/3) 表記法
(Formal semantics of LTL: Notation)

パス $\pi = (s_0 s_1 \cdots)$

A path is represented by an infinite sequence $\pi=(s_0 s_1 s_2 \ldots)$ of states.

$\pi^i = (s_i s_{i+1} \cdots)$: 第 i 番目以降の部分パス

A subpath of π starting from s_i , i.e. $(s_i s_{i+1} \ldots)$, is denoted by π^i.

$\pi \models f$: LTL 式 f がパス π において真(π は f を満たす)

The LTL formula f is true for the path π (π fulfills f).

$\pi \not\models f$: 上式の否定

The negation of the above formula

LTLの形式的意味論(2/3)　時相演算子を含まないとき

(Formal semantics of LTL: When no temporal operators involved)

LTL 式の真偽値の帰納的な定義

($p \in AP$ は原始命題，f, g は LTL 式)

$$\pi \models p \quad \Leftrightarrow \quad p \in L(s_0)$$

$$\pi \models \neg f \quad \Leftrightarrow \quad \pi \not\models f$$

$$\pi \models f \vee g \quad \Leftrightarrow \quad \pi \models f \quad or \quad \pi \models g$$

$$\pi \models f \wedge g \quad \Leftrightarrow \quad \pi \models f \quad and \quad \pi \models g$$

$$\pi \models f \rightarrow g \quad \Leftrightarrow \quad \pi \models \neg f \vee g$$

The inductive definition of the truth for LTL formulas, where p is an atomic proposition and f, g are LTL formulas.

s_0に付けられた
ラベル（原始命題の集合）

the label (a subset of AP) attached to s_0

f が時相演算子を含まないとき，

f の真偽値はパス π の最初の状態 s_0 における真偽値となる.

If f involves no temporal operators,
the truth of f is its truth in the initial state s_0 of the path π.

LTLの形式的意味論(3/3)　時相演算子を含むとき

(Formal semantics of LTL: When temporal operators involved)

$$\pi=(s_0 s_1 s_2 \ldots)$$

$$\pi^i=(s_i s_{i+1} \ldots)$$

$$\pi \models \mathbf{X}f \quad \Leftrightarrow \quad \pi^1 \models f$$

$$\pi \models \mathbf{F}f \quad \Leftrightarrow \quad \text{there exists a } k \geq 0 \text{ such that } \pi^k \models f$$

$$\pi \models \mathbf{G}f \quad \Leftrightarrow \quad \text{for all } i \geq 0,\ \pi^i \models f$$

$$\pi \models f\,\mathbf{U}g \quad \Leftrightarrow \quad \text{there exists a } k \geq 0 \text{ such that } \pi^k \models g \text{ and}$$

$$\text{for all } 0 \leq j < k,\ \pi^j \models f$$

CTLの形式的意味論(1/3)　表記法

(Formal semantics of CTL: Notation)

$s \models f$ ：　CTL 式 f が, 状態 s から始まる計算木において真

（s は f を満たす）

The CTL formula f is true for the computation tree starting from the state s (s fulfills f).

$s \not\models f$ ：　上式の否定 (The negation of the above formula)

CTLの形式的意味論(2/3)　時相演算子を含まないとき

(Formal semantics of CTL: When no temporal operators involved)

CTL 式の真偽値の帰納的な定義

($p \in AP$ は原始命題，f, g は CTL 式)

$$
\begin{aligned}
s \models p &\Leftrightarrow p \in L(s) \\
s \models \neg f &\Leftrightarrow s \not\models f \\
s \models f \vee g &\Leftrightarrow s \models f \quad or \quad s \models g \\
s \models f \wedge g &\Leftrightarrow s \models f \quad and \quad s \models g \\
s \models f \rightarrow g &\Leftrightarrow s \models \neg f \vee g
\end{aligned}
$$

The inductive definition of the truth for CTL formulas, where p is an atomic proposition and f, g are CTL formulas.

s に付けられた
ラベル（原始命題の集合）

the label (a set of atomic propositions) attached to s

f が時相演算子を含まないとき，

f の真偽値は状態 s (計算木の根)における真偽値となる.

If f involves no temporal operators,
the truth of f is its truth in the state s , the root of the computation tree.

CTLの形式的意味論(3/3)　時相演算子を含むとき

(Formal semantics of CTL: When temporal operators involved)

$$s \models \mathbf{E}f \Leftrightarrow \text{there exists a path } \pi \text{ starting from } s \text{ such that } \pi \models f$$

$$s \models \mathbf{A}f \Leftrightarrow \text{for every path } \pi \text{ starting from } s,\ \pi \models f$$