Automaten und Formale Sprachen

μ-calculus

Slides from: Tevfik Bultan (UC Santa Barbara)

What is a Fixpoint (aka, Fixed Point)

Given a function

$\mathcal{F} : D \rightarrow D$

$x \in D$ is a fixpoint of $\mathcal{F}$ if and only if $\mathcal{F}(x) = x$

Temporal Properties ≡ Fixpoints

[Emerson and Clarke 80]

Here are some interesting CTL equivalences:

$AG\ p = p \wedge AX\ AG\ p$

$EG\ p = p \wedge EX\ EG\ p$

$AF\ p = p \vee AX\ AF\ p$

$EF\ p = p \vee EX\ EF\ p$

$p\ AU\ q = q \vee (p \wedge AX\ (p\ AU\ q))$

$p\ EU\ q = q \vee (p \wedge EX\ (p\ EU\ q))$

Note that we wrote the CTL temporal operators in terms of themselves and EX and AX operators

Functionals

- Given a transition system T=(S, I, R), we will define functions from sets of states to sets of states
  - $\mathcal{F} : 2^S \rightarrow 2^S$
- For example, one such function is the EX operator (which computes the precondition of a set of states)
  - EX : $2^S \rightarrow 2^S$

  which can be defined as:

  $EX(p) = \{ s \mid (s,s') \in R \text{ and } s' \in p \}$

Abuse of notation: I am using p to denote the set of states which satisfy the property p (i.e., the truth set of p)

Functionals

- Now, we can think of all temporal operators also as functions from sets of states to sets of states
- For example:

  $AX\ p = \neg EX(\neg p)$

or if we use the set notation

  $AX\ p = (S - EX(S - p))$

Abuse of notation: I will use the set and logic notations interchangeably.

Logic	*Set*
$p \wedge q$	$p \cap q$
$p \vee q$	$p \cup q$
$\neg p$	$S - p$
False	$\varnothing$
True	S

Lattice

The set of states of the transition system forms a lattice:

• lattice	2^S
• partial order	$\subseteq$
• bottom element	$\emptyset$
• top element	S
• Least upper bound (lub) (aka join) operator	$\cup$
• Greatest lower bound (glb) (aka meet) operator	$\cap$

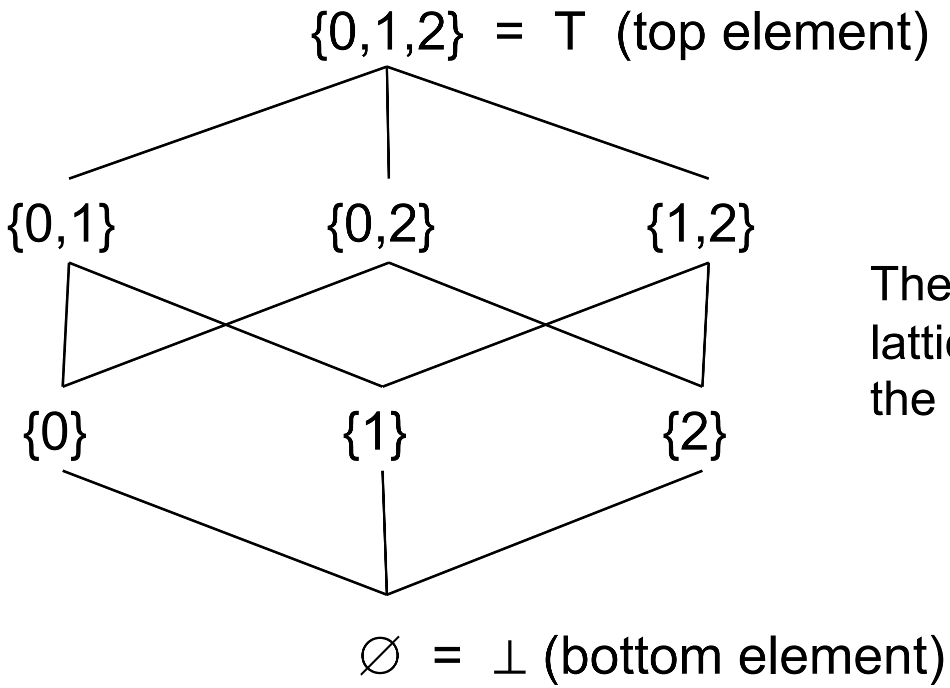
An Example Lattice

{∅, {0}, {1}, {2}, {0,1},{0,2},{1,2},{0,1,2}}

partial order: ⊆ (subset relation)

bottom element: ∅ = ⊥ top element: {0,1,2} = T

lub: ∪ (union) glb: ∩ (intersection)



The Hasse diagram for the example lattice (shows the transitive reduction of the corresponding partial order relation)

Temporal Properties ≡ Fixpoints

Based on the equivalence

$$EF\ p = p \vee EX\ EF\ p$$

we observe that EF p is a fixpoint of the following function:

$$\mathcal{F}\ y = p \vee EX\ y$$

$$\mathcal{F}\ (EF\ p) = EF\ p$$

In fact, EF p is the least fixpoint of $\mathcal{F}$, which is written as:

$$EF\ p = \mu\ y\ .\ \mathcal{F}\ y = \mu\ y\ .\ p \vee EX\ y$$

(μ means least fixpoint)

Temporal Properties ≡ Fixpoints

Based on the equivalence

$$EG\ p = p \wedge AX\ EG\ p$$

we observe that EG p is a fixpoint of the following function:

$$\mathcal{F}\ y = p \wedge EX\ y$$

$$\mathcal{F}\ (EG\ p) = EG\ p$$

In fact, EG p is the greatest fixpoint of $\mathcal{F}$, which is written as:

$$EG\ p = \nu\ y\ .\ \mathcal{F}\ y = \nu\ y\ .\ p \wedge EX\ y$$

(ν means greatest fixpoint)

Fixpoint Characterizations

Fixpoint Characterization	Equivalences
$AG\ p = \nu y\ .\ p \wedge AX\ y$	$AG\ p = p \wedge AX\ AG\ p$
$EG\ p = \nu y\ .\ p \wedge EX\ y$	$EG\ p = p \wedge EX\ EG\ p$
$AF\ p = \mu y\ .\ p \vee AX\ y$	$AF\ p = p \vee AX\ AF\ p$
$EF\ p = \mu y\ .\ p \vee EX\ y$	$EF\ p = p \vee EX\ EF\ p$
$p\ AU\ q = \mu y\ .\ q \vee (p \wedge AX\ (y))$	$p\ AU\ q = q \vee (p \wedge AX\ (p\ AU\ q))$
$p\ EU\ q = \mu y\ .\ q \vee (p \wedge EX\ (y))$	$p\ EU\ q = q \vee (p \wedge EX\ (p\ EU\ q))$

Least Fixpoint

Given a monotonic function $\mathcal{F}$, its least fixpoint is the greatest lower bound (glb) of all the reductive elements :

$$\mu\, y \,.\, \mathcal{F}\, y = \cap \{\, y \mid \mathcal{F}\, y \subseteq y \,\}$$

The least fixpoint $\mu\, y \,.\, \mathcal{F}\, y$ is the limit of the following sequence (assuming $\mathcal{F}$ is $\cup$-continuous):

$$\emptyset,\ \mathcal{F}\,\emptyset,\ \mathcal{F}^2\,\emptyset,\ \mathcal{F}^3\,\emptyset,\ \ldots$$

If S is finite, then we can compute the least fixpoint using the above sequence

EF Fixpoint Computation

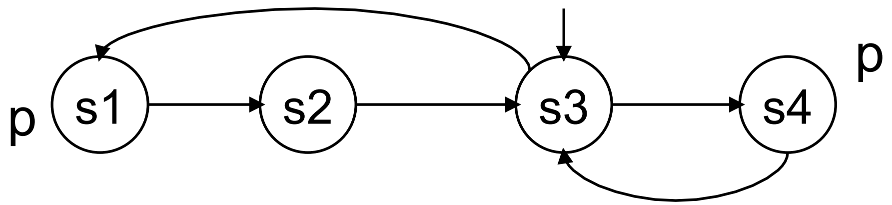
EF p = μ y . p $\vee$ EX y is the limit of the sequence:

$\emptyset$, p$\vee$EX $\emptyset$, p$\vee$EX(p$\vee$EX $\emptyset$) , p$\vee$EX(p$\vee$EX(p$\vee$ EX $\emptyset$)) , ...

which is equivalent to

$\emptyset$, p, p $\vee$ EX p , p $\vee$ EX (p $\vee$ EX (p)) , ...

EF Fixpoint Computation



Start

$\emptyset$

1st iteration

$p \vee EX\ \emptyset = \{s1,s4\} \cup EX(\emptyset) = \{s1,s4\} \cup \emptyset = \{s1,s4\}$

2nd iteration

$p \vee EX(p \vee EX\ \emptyset) = \{s1,s4\} \cup EX(\{s1,s4\}) = \{s1,s4\} \cup \{s3\} = \{s1,s3,s4\}$

3rd iteration

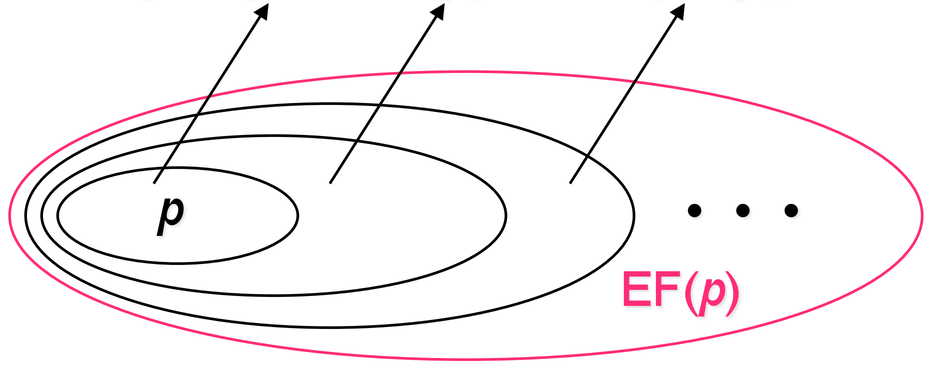
$p \vee EX(p \vee EX(p \vee EX\ \emptyset)) = \{s1,s4\} \cup EX(\{s1,s3,s4\}) = \{s1,s4\} \cup \{s2,s3,s4\} = \{s1,s2,s3,s4\}$

4th iteration

$p \vee EX(p \vee EX(p \vee EX(p \vee EX\ \emptyset))) = \{s1,s4\} \cup EX(\{s1,s2,s3,s4\}) = \{s1,s4\} \cup \{s1,s2,s3,s4\}$
$= \{s1,s2,s3,s4\}$

EF Fixpoint Computation

EF(p) ≡ states that can reach p ≡ $p \cup$ EX(p) $\cup$ EX(EX(p)) $\cup$ …



Greatest Fixpoint

Given a monotonic function $\mathcal{F}$, its greatest fixpoint is the least upper bound (lub) of all the extensive elements:

$$\nu\, y.\ \mathcal{F}\, y = \cup \{ y \mid \mathcal{F}\, y \subseteq y \}$$

The greatest fixpoint $\nu\, y\, .\ \mathcal{F}\, y$ is the limit of the following sequence (assuming $\mathcal{F}$ is $\cap$-continuous):

$$S,\ \mathcal{F}\, S,\ \mathcal{F}^2\, S,\ \mathcal{F}^3\, S,\ \ldots$$

If S is finite, then we can compute the greatest fixpoint using the above sequence

EG Fixpoint Computation

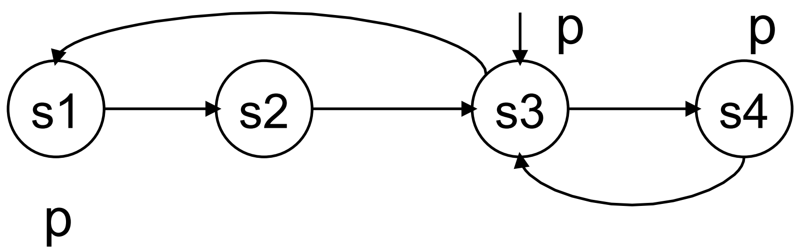
Similarly, $EG\ p = \nu y\ .\ p \wedge EX\ y$ is the limit of the sequence:

$S,\ p \wedge EX\ S,\ p \wedge EX(p \wedge EX\ S)\ ,\ p \wedge EX(p \wedge EX\ (p \wedge EX\ S))\ ,\ ...$

which is equivalent to

$S,\ p,\ p \wedge EX\ p\ ,\ p \wedge EX\ (p \wedge EX\ (p)\ )\ ,\ ...$

EG Fixpoint Computation



Start
$S = \{s1,s2,s3,s4\}$

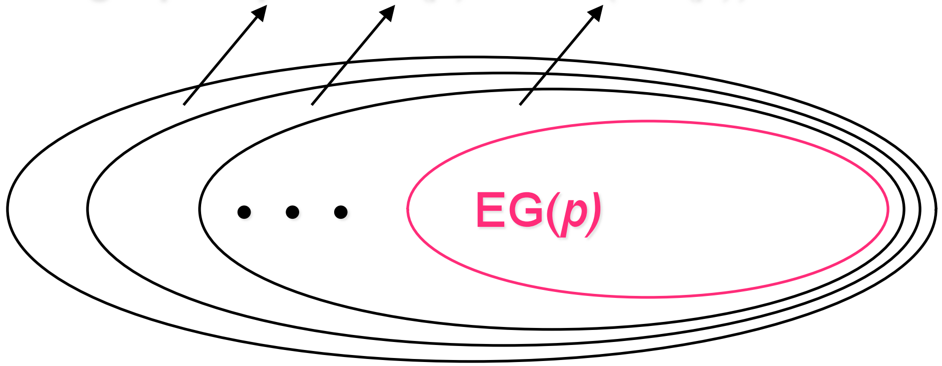
1st iteration
$p \wedge EX\ S = \{s1,s3,s4\} \cap EX(\{s1,s2,s3,s4\}) = \{s1,s3,s4\} \cap \{s1,s2,s3,s4\} = \{s1,s3,s4\}$

2nd iteration
$p \wedge EX(p \wedge EX\ S) = \{s1,s3,s4\} \cap EX(\{s1,s3,s4\}) = \{s1,s3,s4\} \cap \{s2,s3,s4\} = \{s3,s4\}$

3rd iteration
$p \wedge EX(p \wedge EX(p \wedge EX\ S)) = \{s1,s3,s4\} \cap EX(\{s3,s4\}) = \{s1,s3,s4\} \cap \{s2,s3,s4\} = \{s3,s4\}$

EG Fixpoint Computation

EG(p) ≡ states that can avoid reaching $\neg p$ ≡ $p \cap$ EX(p) $\cap$ EX(EX(p)) $\cap$ …



μ-Calculus

μ-Calculus is a temporal logic which consist of the following:

- Atomic properties AP
- Boolean connectives: $\neg$, $\wedge$, $\vee$
- Precondition operator: EX
- Least and greatest fixpoint operators: $\mu\, y \,.\, \mathcal{F}\, y$ and $\nu\, y.\, \mathcal{F}\, y$
  - $\mathcal{F}$ must be syntactically monotone in y
    - meaning that all occurrences of y in within $\mathcal{F}$ fall under an even number of negations

μ-Calculus

- μ-calculus is a powerful logic
  - Any CTL* property can be expressed in μ-calculus
- So, if you build a model checker for μ-calculus you would handle all the temporal logics we discussed: LTL, CTL, CTL*
- One can write a μ-calculus model checker using the basic ideas about fixpoint computations that we discussed
  - However, there is one complication
    - Nested fixpoints!