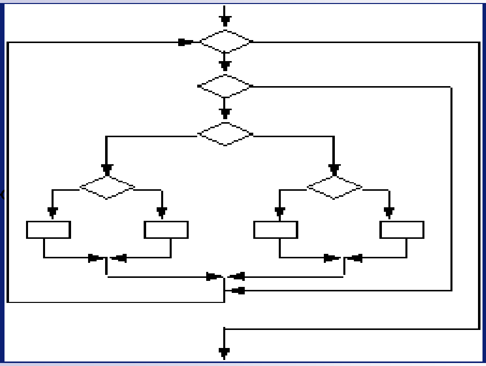
Spring 2007



Program correctness

SAT and its correctness

Marcello Bonsangue

Leiden Institute of Advanced Computer Science
Research & Education

Context

1. We have defined the semantics of CTL formulas $M,s \models \phi$
2. We have given an efficient method for model checking a CTL formula returning all states s such that $M,s \models \phi$

Next we present an algorithm for it and proves its correctness

The algorithm SAT

- SAT stands for 'satisfies'
  - Input: a well-formed CTL formula
  - Output: a subset of the states of a transition system M = <S, $\rightarrow$,l>

- Written in Pascal-like
  - function return
  - local_var
  - while do od
  - case is end_case

The main function (I)

<u>function</u> SAT(ϕ)

<u>begin</u>

<u>case</u> ϕ <u>is</u>

T : <u>return</u> S

$\bot$: <u>return</u> $\varnothing$

atomic : <u>return</u> $\{s \in S \mid \phi \in l(s)\}$

$\neg\phi_1$: <u>return</u> S - SAT(ϕ_1)

$\phi_1 \wedge \phi_2$: <u>return</u> SAT(ϕ_1) $\cap$ SAT(ϕ_2)

$\phi_1 \vee \phi_2$: <u>return</u> SAT(ϕ_1) $\cup$ SAT(ϕ_2)

$\phi_1 \Rightarrow \phi_2$: <u>return</u> SAT($\neg\phi_1 \vee \phi_2$)

⋮

The main function (II)

⋮

$AX\phi_1$: return SAT($\neg EX \neg \phi_1$)

$EX\phi_1$: return SAT_EX(ϕ_1)

$A[\phi_1 \, U \, \phi_2]$: return

SAT($\neg E[\neg \phi_2 U(\neg \phi_1 \wedge \neg \phi_2)] \vee EG \neg \phi_2$)

$E[\phi_1 \, U \, \phi_2]$: return SAT_EU(ϕ_1, ϕ_2)

$EF\phi_1$: return SAT($E[T \, U \, \phi_1]$)

$AF\phi_1$: return SAT_AF(ϕ_1)

$EG\phi_1$: return SAT($\neg AF \neg \phi_1$) /*SAT_EG(ϕ_1)*/

$AG\phi_1$: return SAT($\neg EF \neg \phi_1$)

end_case

end

The function SAT_EX

<u>function</u> SAT_EX(ϕ)

<u>local_var</u> X,Y

<u>begin</u>

X := SAT(ϕ)

Y := $\{ s \in S \mid \exists s \rightarrow s' : s' \in X \}$

<u>return</u> Y

<u>end</u>

The function SAT_AF

```
function SAT_AF(ϕ)
local_var X,Y
begin
    X := S
    Y := SAT(ϕ)
    while X ≠ Y do
        X := Y
        Y := Y ∪ { s ∈ S | ∀s → s' : s' ∈ Y }
    od
    return Y
end
```

The function SAT_EU

<u>function</u> SAT_EU(ϕ,ψ)

<u>local_var</u> W,X,Y

<u>begin</u>

W := SAT(ϕ)
X := S
Y := SAT(ψ) /* Calculated only once */

<u>while</u> X $\neq$ Y <u>do</u>

X := Y

Y := Y $\cup$ (W $\cap$ { s $\in$ S | $\exists$s $\rightarrow$ s' : s' $\in$ Y })

<u>od</u>

<u>return</u> Y

<u>end</u>

The function SAT_EG

<u>function</u> SAT_EG(ϕ)

<u>local_var</u> X,Y

<u>begin</u>

X := $\varnothing$
Y := SAT(ϕ)

<u>while</u> X $\neq$ Y <u>do</u>

X := Y

Y := Y $\cap$ $\{ s \in S \mid \exists s \rightarrow s' : s' \in Y \}$

<u>od</u>

<u>return</u> Y

<u>end</u>

Does it work?

- Claim: For a given model $M=\langle S, \rightarrow, I\rangle$ and well-formed CTL formula ϕ,

$$SAT(\phi) = \{ s \in S \mid M,s \models \phi \} \stackrel{def}{=} [[\phi]]$$

Is this true?

The proof (I)

- The claim is proved by induction on the structure of the formula.
- For ϕ = T, $\bot$, or atomic the set $[[\phi]]$ is computed directly
- For $\neg\phi$, $\phi_1 \wedge \phi_2$, $\phi_1 \vee \phi_2$ or $\phi_1 \Rightarrow \phi_2$ we apply induction and predicate logic equivalences
  - Example:

$$SAT(\phi_1 \vee \phi_2) = SAT(\phi_1) \cup SAT(\phi_2)$$

$$= [[\phi_1]] \cup [[\phi_2]] \qquad \text{(induction)}$$

$$= [[\phi_1 \vee \phi_2]]$$

The proof (II)

- For EXϕ we apply induction

$$
\begin{aligned}
SAT(EX\phi) &= SAT\_EX(\phi) \\
&= \{ s \in S \mid \exists\, s \to s' : s' \in SAT(\phi)\} \\
&= \{ s \in S \mid \exists s \to s' : s' \in [[\phi]]\} && \text{(induction)} \\
&= \{ s \in S \mid \exists s \to s' : M,s' \vDash \phi\} && \text{(definition [[-]])} \\
&= \{ s \in S \mid M,s \vDash EX\phi\} && \text{(definition } \vDash \text{)} \\
&= [[EX\phi]] && \text{(definition [[-]])}
\end{aligned}
$$

The proof (III)

- For $AX\phi$, $A[\phi_1 \text{ U } \phi_2]$, $EF\phi$, or $AG\phi$ we can rely on logical equivalences and on the correctness of SAT_EX, SAT_AF, SAT_EU, and SAT_EG
  - Example:

    $SAT(AX\phi) = SAT(\neg EX\neg\phi)$

    $= S - SAT\_EX(\neg\phi)$ (def. $SAT(\neg\phi)$)

    $= S - [[EX\neg\phi]]$ (correctness SAT_EX)

    $= [[AX\phi]]$ (logical equivalence)

But we still have to prove the correctness of SAT_AF, SAT_EU, and SAT_EG

EG as fixed point

Recall that $EG\phi \equiv \phi \wedge EX\ EG\phi$. Since

$$EX\psi = \{ s \in S \mid \exists\ s \rightarrow s' : s' \in [[\psi]]\}$$

we have the following fixed-point definition of EG

$$[[EG\phi]] = [[\phi]] \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in [[EG\phi]]\}$$

Fixed points

- Let S be a set and F:Pow(S) $\rightarrow$ Pow(S) be a a function
  - F is monotone if

    $$X \subseteq Y \text{ implies } F(X) \subseteq F(Y)$$

    for all subsets X and Y of S
  - A subset X of S is a fixed point of F if

    $$F(X) = X$$
  - A subset X of S is a least fixed point of F if

    $$F(X) = X \text{ and } X \subseteq Y$$

    for all fixed point Y of F

Examples

- S = {s,t} and $F: X \mapsto X \cup \{s\}$
  - F is monotone
  - {s} and {s,t} are all fixed points of F
  - {s} is the least fixed point of F

- S = {s,t} and G:X↦<u>if</u> X={s} <u>then</u> {t} <u>else</u> {s}
  - G is not monotone
    - $\{s\} \subseteq \{s,t\}$ but $G(\{s\}) = \{t\} \not\subset \{s\} = G(\{s,t\})$
  - G does not have any fixed point

Fixed points (II)

Let $F^i(X) = \underbrace{F(F(\ldots F}_{\text{i-times}}(X)\ldots))$ for $i > 0$ (thus $F^1(X) = F(X)$)

- Theorem: Let S be a set with n+1 elements. If $F:Pow(S) \rightarrow Pow(S)$ is a monotone function then
  1) $F^{n+1}(\emptyset)$ is the least fixed point of F
  2) $F^{n+1}(S)$ is the greatest fixed point of F

⇓

Least and greatest fixed points can be computed and the computation is guaranteed to terminate ! ☺

Computing EGϕ

- To find a set [[EGϕ]] such that

$$[[EG\phi]] = [[\phi]] \cap \{ s \in S \mid \exists s \to s' : s' \in [[EG\phi]]\}$$

we look if it is a fixed point of the function

$$F(X) = [[\phi]] \cap \{ s \in S \mid \exists s \to s' : s' \in X\}$$

- Theorem: Let n = |S| be the size of S and F defined as above. We have
  1. F is monotone
  2. [[EGϕ]] is the greatest fixed point of F
  3. $[[EG\phi]] = F^{n+1}(S)$

Correctness of SAT_EG

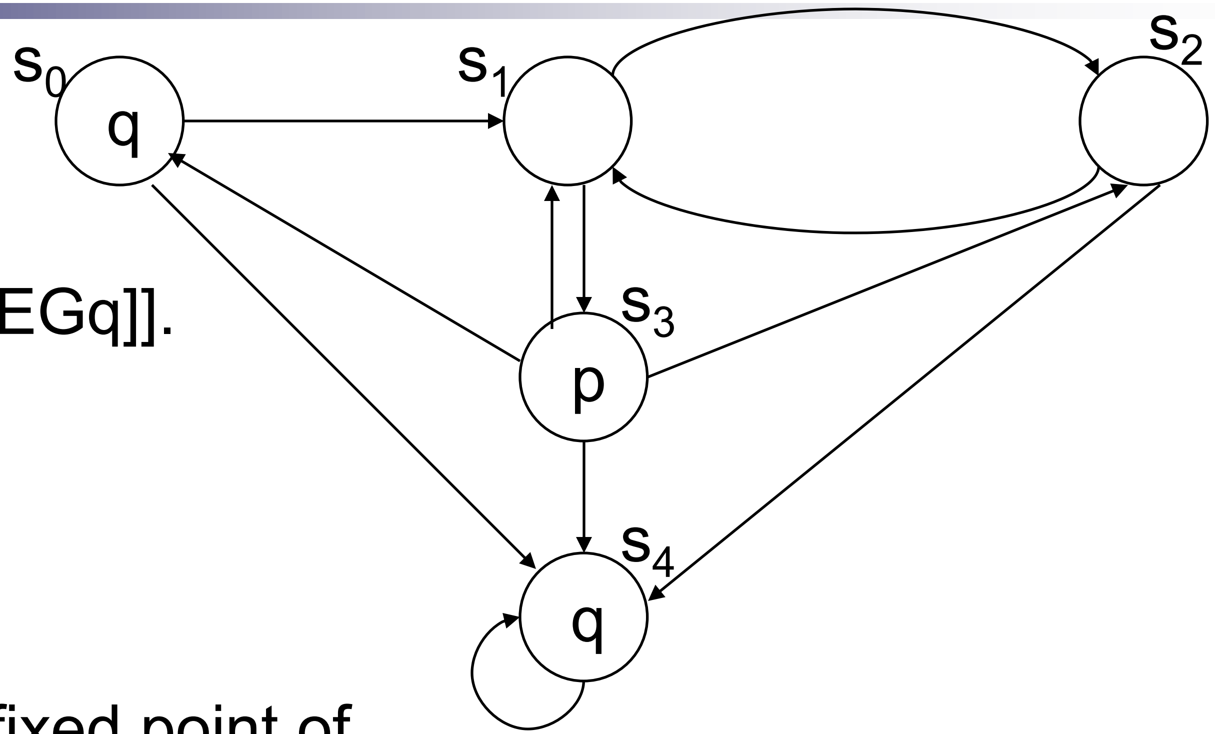
1. Inside the loop it always holds $Y \subseteq SAT(\phi)$
2. Because $Y \subseteq SAT(\phi)$, substitute in SAT_EG

   $Y := Y \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in Y \}$

   with $Y :=$ $SAT(\phi)$ $\cap \{ s \in S \mid \exists s \rightarrow s' : s' \in Y \}$
3. Note that SAT_EG(ϕ) is calculating the greatest fixed point (use induction!)

   $$F(X) = [[\phi]] \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in X\}$$
4. It follows from the previous theorem that SAT_EG(ϕ) terminates and computes $[[EG\phi]]$.

Example: EG



Let us compute [[EGq]].

It is the greatest fixed point of

$$F(X) = [[q]] \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in X \}$$
$$= \{s_0, s_4\} \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in X \}$$

Example: EG

- Iterating F on S until it stabilizes
  - $F^1(S) = \{s_0, s_4\} \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in S \}$

    $= \{s_0, s_4\} \cap S$

    $= \{s_0, s_4\}$

  - $F^2(S) = F(F^1(S))$

    $= F(\{s_0, s_4\})$

    $= \{s_0, s_4\} \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in \{s_0, s_4\} \}$

    $= \{s_0, s_4\}$

- Thus $\{s_0, s_4\}$ is the greatest fixed point of F and equals [[EGq]]

EU as fixed point

- Recall that $E[\phi \; U \; \psi] \equiv \psi \vee (\phi \wedge EX \; E[\phi \; U \; \psi])$.
- Since $EX\varphi = \{ s \in S \mid \exists \; s \rightarrow s' : s' \in [[\varphi]]\}$ we obtain

$$[[E[\phi U \psi]]] = [[\psi]] \cup ([[\phi]] \cap \{s \in S \mid \exists s \rightarrow s' : s' \in [[E[\phi U \psi]]]\})$$

Computing $E[\phi \text{ U } \psi]$

- As before, we show that $[[E[\phi \text{ U } \psi]]]$ is a fixed point of the function

  $$G(X) = [[\psi]] \cup ([[\phi]] \cap \{ s \in S \mid \exists s \to s' : s' \in X\})$$

- Theorem: Let $n = |S|$ be the size of S and G defined as above. We have
  1. G is monotone
  2. $[[E[\phi \text{ U } \psi]]]$ is the least fixed point of G
  3. $[[E[\phi \text{ U } \psi]]] = G^{n+1}(\emptyset)$

Correctness of SAT_EU

1. Inside the loop it always holds $W=SAT(\phi)$ and $Y \supseteq SAT(\psi)$.

2. Substitute in SAT_EU

$$Y:=Y \cup (W \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in Y \})$$

with

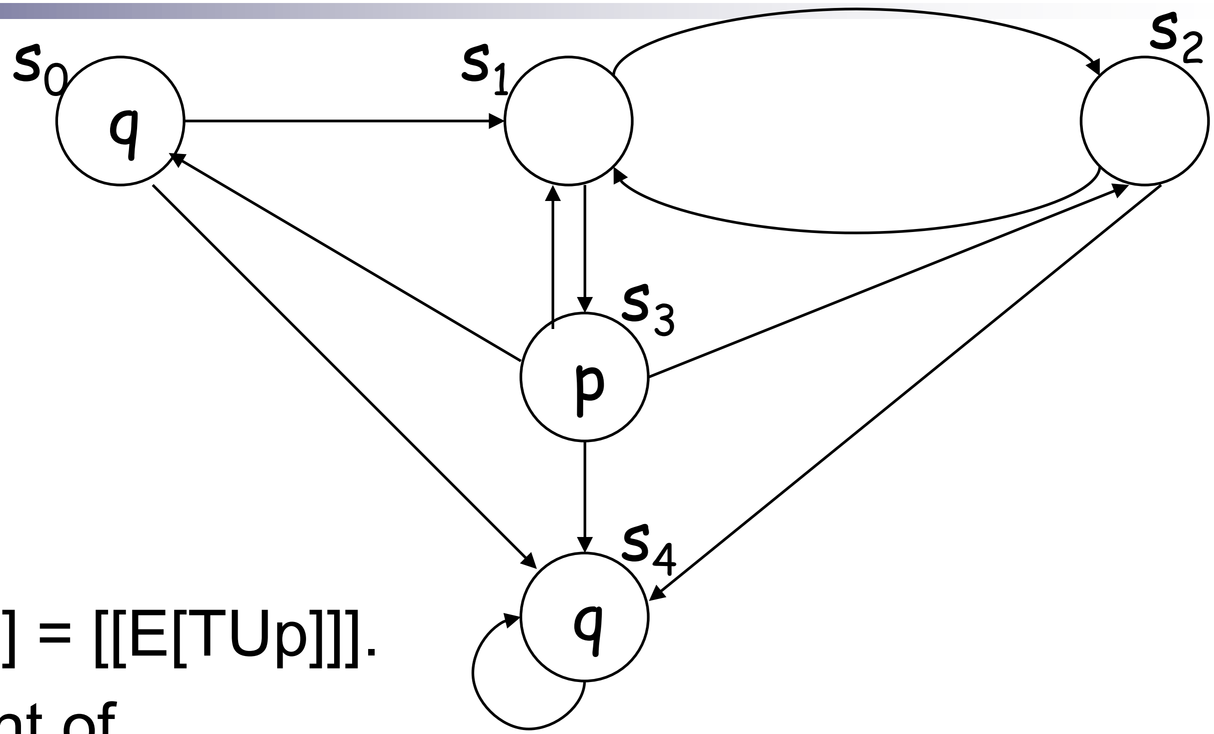
$$Y:=SAT(\psi) \cup (SAT(\phi) \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in Y \})$$

3. Note that $SAT\_EU(\phi)$ is calculating the least fixed point of

$$G(X) = [[\psi]] \cup ([[\phi]] \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in X\})$$

4. It follows from the previous theorem that $SAT\_EU(\phi,\psi)$ terminates and computes $[[E[\phi U \psi]]]$

Example: EU



Let us compute [[EFp]] = [[E[TUp]]].

It is the least fixed point of

$$G(X) = [[p]] \cup ([[T]] \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in X\})$$
$$= \{s_3\} \cup (S \cap \{ s \in S \mid \exists s \rightarrow s' : s' \in X \})$$
$$= \{s_3\} \cup \{ s \in S \mid \exists s \rightarrow s' : s' \in X \}$$

Example: EU

- Iterating G on $\emptyset$ until it stabilizes we have
  - $G^1(\emptyset) = \{s_3\} \cup \{ s \in S \mid \exists s \to s' : s' \in \emptyset \}$
    $= \{s_3\} \cup \emptyset = \{s_3\}$
  - $G^2(\emptyset) = G(G^1(\emptyset)) = G(\{s_3\})$
    $= \{s_3\} \cup \{ s \in S \mid \exists s \to s' : s' \in \{s_3\} \}$
    $= \{s_1,s_3\}$
  - $G^3(\emptyset) = G(G^2(\emptyset)) = G(\{s_1,s_3\})$
    $= \{s_3\} \cup \{ s \in S \mid \exists s \to s' : s' \in \{s_1,s_3\} \}$
    $= \{s_0,s_1, s_2,s_3\}$
  - $G^4(\emptyset) = G(G^3(\emptyset)) = G(\{s_0,s_1, s_2,s_3\})$
    $= \{s_3\} \cup \{ s \in S \mid \exists s \to s' : s' \in \{s_0,s_1, s_2,s_3\} \}$
    $= \{s_0,s_1, s_2,s_3\}$
- Thus [[EFp]] = [[E[TUp]]] $= \{s_0,s_1,s_2,s_3\}$.

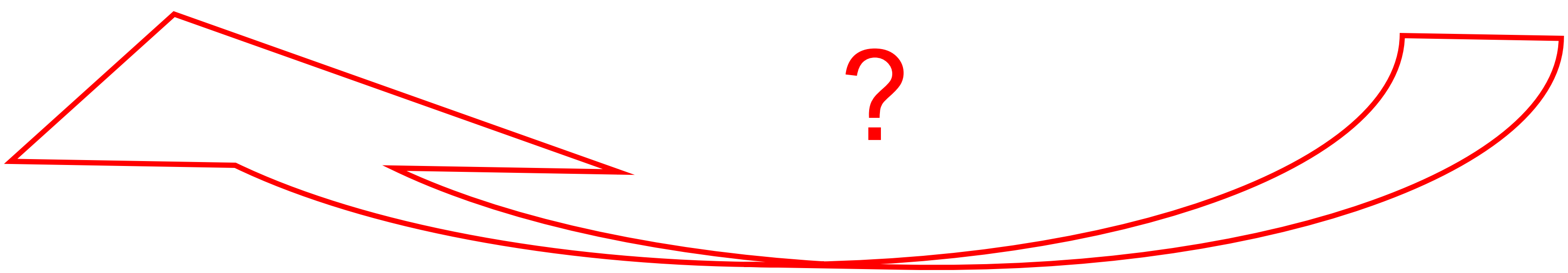
AF as fixed point

Since $AF\phi \equiv \phi \vee AX\ AF\phi$ and

$$AX\varphi = \{ s \in S \mid \forall s \to s' : s' \in [[\varphi]]\}$$

we obtain

$$[[AF\phi]] = [[\phi]] \cup \{ s \in S \mid \forall s \to s' : s' \in [[AF\phi]]\}$$



Computing AFϕ

- Again, consider $[[AF\phi]]$ as a fixed point of the function

$$H(X) = [[\phi]] \cup \{ s \in S \mid \forall s \rightarrow s' : s' \in X\}$$

- Theorem: Let n = |S| be the size of S and G defined as above. We have
  1. H is monotone
  2. $[[AF\phi]]$ is the least fixed point of H
  3. $[[AF\phi]] = H^{n+1}(\emptyset)$

Correctness of SAT_AF

1. Inside the loop it always holds $Y \supseteq SAT(\phi)$.

2. Substitute in SAT_AF

$$Y := Y \cup \{ s \in S \mid \forall s \to s' : s' \in Y \})$$

with

$$Y := SAT(\phi) \cup \{ s \in S \mid \forall s \to s' : s' \in Y \}$$

3. Note that SAT_AF(ϕ) is calculating the least fixed point of

$$H(X) = [[\phi]] \cup \{ s \in S \mid \forall s \to s' : s' \in X\}$$

4. It follows from the previous theorem that AT_AF(ϕ) terminates and computes [[**AF**ϕ]]