Software Verification
Exercise class:
Real Time Systems

Carlo A. Furia
Exercises:
Does the property hold?
Does the property hold?

\[
\begin{align*}
x &:= 0 & x &:= 0 \\
x &= 1 & x &= 1 \\
a & & a
\end{align*}
\]
Does the property hold?

Yes:
- it simply means that \( a \) holds at every position in the word (if any)
Does the property hold?

\[ \text{S1} \]

\[
\begin{align*}
x & := 0 \\
x & := 0 \\
x & = 1 \\
x & = 1
\end{align*}
\]

\[ \text{S2} \]

\[ [] \ (<>=1 \ a ) \]
Does the property hold?

\[ \text{No:} \]

- this requires that there is always a future position, 1 time unit in the future, where \( a \) holds
- but this is not the case in the last position of any (non-empty) timed word
Does the property hold?

\[
\begin{align*}
\text{S1} & : x := 0 \quad x := 0 \\
\text{S2} & : x = 1 \quad x = 1 \\
\end{align*}
\]

\[
[]([[]]=1 \ a)
\]
Does the property hold?

Yes:  
- the formula just requires that there if there is a future position 1 time unit in the future, then a holds there  
- the automaton accepts only a's every time unit, hence the property is satisfied by any word accepted by the automaton

$$
[] ( []=1 \ a )
$$
Does the property hold?

\[
\begin{align*}
\text{S1:} & \quad x, y := 0 \\
& \quad 0 < x, y < 1 \\
\text{S2:} & \quad x := 0 \\
\text{S3:} & \quad y := 0
\end{align*}
\]

\[\Box (a \Rightarrow \leftrightarrow (0,1) \ c)\]
Does the property hold?

\[
[a \Rightarrow \lll (0,1) \ c]
\]

Yes:
- clock \( x \) is reset upon reading \( a \)
- after that, it is checked upon reading \( c \)
- the constraint requires that \( x \) is in the range \((0,1)\)
Does the property hold?

\[ [] \ (a \Rightarrow \langle (0,1) \ b \rangle) \]
Does the property hold?

\[\Box (a \Rightarrow \llbracket 0,1 \rrbracket b)\]

Yes:

- Clock \( x \) is reset upon reading \( a \); after that, it is checked upon reading \( c \), which is always preceded by a reading of \( b \).
- If \( b \) occurs later than or exactly after 1 time unit since the reading of \( b \), the same occurs for the reading of \( c \).
- In this case, the constraint on \( x \) would be violated.
Does the property hold?

\[ [\alpha \Rightarrow (\alpha \lor \beta) \cup (0,1) \ c] \]
Does the property hold?

\[ [](a \Rightarrow (a \lor b) \cup (0,1) c) \]

Yes:
- clock x is reset upon reading a
- after that there is one reading of b followed by a reading of c, which satisfies the sequence of events required by the until formula
- as far as timing is concerned, c must occur within interval of time (0,1) since a occurred because of the clock constraint 0 < x, y < 1
Does the property hold?

\[
[] (a \Rightarrow (a \lor b) \cup (1,2) c)
\]
Does the property hold?

\[ \text{No:} \]

- if the “next” \( c \) is considered w.r.t when \( a \) occurs, it cannot happen in interval \((1,2)\)
- if a successive occurrence of \( c \) is considered, it is preceded by at least another occurrence of \( c \), which is not admitted by \( a \lor b \)
Exercises:
Region automaton construction
Build the region automaton for:
Build the region automaton for:

\[ x := 0 \]
\[ x = 1 \]

\[ x := 0 \]
\[ x = 1 \]
Build the region automaton for:

S1

\[ x, y := 0 \]

\[ 0 < x, y < 1 \]

S2

\[ x := 0 \]

S3

\[ y := 0 \]
Build the region automaton for:

S1: \( x = y = 0 \)

S2: 
- \( x = 0 \) (\( 0 < y < 1 \))
- \( y = 1 \)
- \( y > 1 \)

S3: 
- \( y = 0 \) (\( 0 < x < 1 \))
- \( x = 1 \)
- \( x > 1 \)

Transitions:
- a from S1 to S2
- b from S2 to S3
- c from S3 back to S1

States:
- S1: \( x = y = 0 \)
- S2: Various conditions on x and y
- S3: Various conditions on x and y

Transitions:
- a: \( x = 0 \) to \( 0 < y < 1 \)
- b: \( y = 1 \) to \( y > 1 \)
- c: \( y = 0 \) to \( 0 < x < 1 \)
Build the region automaton for:

Example from: Alur & Dill, 1994
Build the region automaton for:

Example from: Alur & Dill, 1994
Exercises:
Semantics of derived operators
Prove that the satisfaction relation

\[ w, i \models []\langle a, b \rangle F \]

for bounded always, defined as:

\[ []\langle a, b \rangle F \triangleq \neg (\text{True} \lor \langle a, b \rangle \neg F) \]

is equivalent to:

for all \( i \leq j \leq n \) such that \( t(j) - t(i) \in \langle a, b \rangle \) it is: \( w, j \models F \)
MTL derived operators: always

\[ w, i \models [\langle a, b \rangle] F \]

iff

\[ w, i \not\models \neg (\text{True} \mathcal{U} \langle a, b \rangle \neg F) \quad \text{(definition of bounded always)} \]

iff

it is not the case that:
for some \( i \leq j \leq n \) such that \( t(j) - t(i) \in \langle a, b \rangle \) it is: \( w, j \not\models \neg F \)
and for all \( i \leq k < j \) it is \( w, k \models \text{True} \)

(definition of bounded until)

iff

for all \( i \leq j \leq n \) such that \( t(j) - t(i) \in \langle a, b \rangle \) it is: \( \neg w, j \not\models \neg F \)
or for all \( i \leq k < j \) it is \( w, k \not\models \text{False} \)

(push negation inward)

iff

for all \( i \leq j \leq n \) such that \( t(j) - t(i) \in \langle a, b \rangle \) it is: \( \neg w, j \not\models \neg F \)
(dropping false term in disjunction)

iff

for all \( i \leq j \leq n \) such that \( t(j) - t(i) \in \langle a, b \rangle \) it is: \( w, j \models F \)
(simplification of double negation)
MTL derived operators: $X$ and $X^-$

Compare the semantics of:

\[ X^+ F \triangleq \text{True } U=1 F \]

with the semantics of:

\[ X^- F \triangleq F U>0 \text{ True} \]
Semantic of $X+$

$w, i \not\models X+F$

iff

$w, i \not\models \text{True \ U=1 F}$ \hspace{1cm} (definition of $X+$)

iff

$$\text{for some } i \leq j \leq n \text{ such that } t(j) - t(i) = 1 \text{ it is: } w, j \not\models F$$

and for all $i \leq k < j$ it is $w, k \models \text{True}$

(definition of bounded until)

iff

$$\text{for some } i \leq j \leq n \text{ such that } t(j) = t(i) + 1 \text{ it is: } w, j \not\models F$$

(simplify term)
Semantic of X-

\( w, i \nvDash X \text{-} F \)

iff

\( w, i \nvDash F \text{ U>0 True} \) (definition of X-)

iff

for some \( i \leq j \leq n \) such that \( t(j) - t(i) > 0 \) it is: \( w, j \nvDash True \)
and for all \( i \leq k < j \) it is \( w, k \nvDash F \)

(definition of bounded until)

iff

for some \( i < j \leq n \) it is: \( w, j \nvDash True \) and for all \( i \leq k < j \) it is \( w, k \nvDash F \)
(timestamps are strictly increasing by assumption)

iff

\( i < n \) and \( w, i \nvDash F \)
(take \( j = i+1 \) so that \([i, j) = [i,i])\)
Exercises:
Equivalence of MTL formulas
Is formula:

\[
\[ \langle \langle \langle 0 \rangle \rangle \rangle \text{ True}
\]

satisfied by any timed word?
Is formula satisfied?

Semantics of: \( w \not\models [\virgin] \llll \emptyset \mathrm{True} \)

for all positions \( 1 \leq i \leq n \): \( w,i \not\models \llll \emptyset \mathrm{True} \)

Semantics of: \( w,i \models \llll \emptyset \mathrm{True} \)

for some \( j > i \) it is: \( w,j \models \mathrm{True} \)

i.e.: \( i < n \)

Hence: \( w \not\models [\virgin] \llll \emptyset \mathrm{True} \)

holds only for the empty word!
Comparison of formulas

Is formula:

\[
[] \gg\gg 0 \text{ True}
\]

termed word?
Is formula satisfied?

Semantics of: \( w \models [[] \ggg 0 \text{ True} \]

for all positions \( 1 \leq i \leq n \): \( w,i \models \ggg 0 \text{ True} \)

Semantics of: \( w,i \models \ggg 0 \text{ True} \)

for some \( j \geq i \) it is: \( w,j \models \text{True} \)

i.e.: \( \text{True} \)

because one can always take \( j = i \)

Hence: \( w \models [[] \ggg 0 \text{ True} \)

holds for any word.
Comparison of formulas

Is formula:

\[ \langle\rangle[a,b] \langle\rangle[c,d] \ q \]

equivalent or non-equivalent to:

\[ \langle\rangle[a+c,b+d] \ q \]
Inequivalent formulas

Informal meaning of: $<[a,b]<>[c,d]\ q$

- let $i$ be the current position
- there exist a future position $j > i$ in the word with time in $[a,b]$ relative to $i$ such that:
  - there exist another future position $k > j$ in the word with time in $[c,d]$ relative to $j$, where $q$ holds
- in all, the time at which $q$ holds is in $[a+c, b+d]$ relative to $i$

Informal meaning of: $<[a+c,b+d]\ q$

- let $i$ be the current position
- there exist another future position $k > i$ in the word with time in $[a+c,b+d]$ relative to $i$, where $q$ holds

Hence, for instance: timed word $w = (\{\}, 3) (\{q\}, 3+b+c)$ is such that: $w$ satisfies $<[a+c,b+d]\ q$ but it does not satisfy $<[a,b]<>[c,d]\ q$ because there is no intermediate position between the first and the one where $q$ holds