Three-valued semantics for LTL in RV

- Seminar Advanced Verification Techniques
- 23.01.13
- Frederik Hahne fhahne@mail.upb.de
Runtime Verification

- Lightweight Verification Technique
- Check if a **run** of a system satisfies or violates some correctness property
- Detect violations
- “Between” Model-Checking and testing
  - Checking for certain properties (LTL) → MC
  - Over a concrete run of system → Testing
Comparison to Model-Checking

- Model-Checking
  - All possible executions
  - **Language inclusion problem**
    - Büchi automaton for infinite words
- Runtime Verification
  - One execution
  - **Word problem**, does the word belong to the language that is defined by the LTL formula? 

\[ L(M) \subseteq L(\varphi) \]

\[ \omega \subseteq L(\varphi) \]
Linear-time logic
Temporal Operators $G, U, F, X$
Interesting properties (for MC)
$G x$
Safety properties
$G \text{ (request } \rightarrow \text{ F acknowledgment)}$
Liveness properties
Problems (Classical LTL semantics)

- RV only deals with **finite** but **expanding** traces
- X p on the last state of the trace?

1 2 3 ?

X p ? No next state available (yet)
Problems (Classical LTL semantics)

- G, F on an incomplete state space
- Request/Acknowledgment
- Is every request followed eventually by a corresponding acknowledgment?
- Hard to say when only knowing incomplete state space

\[ G \ p \ ? \]
Outline

- Short introduction to RV
- LTL for RV
- Problems
- 4 Maxims/Requirements
- LTL Semantics
  - 2-valued: FLTL (1)
  - 3-valued: LTL3 (2)
  - 4-valued: RV-LTL (3)
- Conclusion
Maxim 1: $\exists x$: Existential Next

- “X should behave like in classical LTL in many cases as possible” (Bauer 2010)
- $X \, p$ evaluates to true if
  - a) a successor state \textit{exists}
  - b) that \textit{satisfies} $p$

\begin{align*}
\{p\} & \rightarrow \{p\} & \{p\} & \rightarrow \text{?} \\
1 & \rightarrow 2 & 2 & \rightarrow 3
\end{align*}

- $X \, p \models T$
- $X \, p \models F$, because there is no “next state”
Maxim 2: $\neg \equiv \text{Complementation by Negation}$

- The negation of a formula yields to
- Complementary and
- Different truth values

\[ [\omega \models \varphi] = [\omega \models \neg \varphi] \]

and

\[ [\omega \models \varphi] \neq [\omega \models \neg \varphi] \]

Remark:

\[ \neg X \varphi \equiv X \neg \varphi \]

holds in classical LTL
Maxim 2: Combine maxim 1 & 2

- Combining maxim 1 and 2 would lead to
  \[ \neg X \varphi \not\equiv X \neg \varphi \]

- Introduce a strong X and a weak X (= \(\overline{X}\)) to meet maxim 1 & 2
  \[ \neg X \varphi \equiv \overline{X} \neg \varphi \]

\[
\begin{align*}
1 \quad \{p\} \\
2 \quad \{p\} \\
3 \quad \{p\} \\
? \\
X p = \text{false} \quad \text{(strong)} \\
\overline{X} p = \text{true} \quad \text{(weak)}
\end{align*}
\]
Maxim 2: Combine maxim 1 & 2

- Combining maxim 1 and 2 would lead to
  \[ \neg X \varphi \neq X \neg \varphi \]

- Introduce a strong X and a weak X (= \(\overline{X}\)) to meet maxim 1 & 2
  \[ \neg X \varphi \equiv \overline{X} \neg \varphi \]

\[ \{p\} \quad \{p\} \quad \{p\} \]

\[ 1 \quad 2 \quad 3 \quad ? \]

\[ X p = \text{false} \quad \text{(strong)} \]
\[ \overline{X} p = \text{true} \quad \text{(weak)} \]
Maxim 3: Sound

- An LTL semantics for RV should not evaluate to true or false prematurely.
- No 2-valued semantics can fulfill this.

{p} \{p\} \{p\}

1 2 3

May evaluate to true or false Depending what happens in the future. We don't know if the property is fulfilled or not!
Maxim 4: Precise

- Be as anticipatory as possible
- Evaluate to true (false) as early as possible
- If the result is determined by the current prefix evaluate to this value

Every possible continuation leads to **false**. An anticipatory semantics would evaluate already to false at this state!

Every possible continuation (still) leads to **false**
Outline

- Short introduction to RV
- LTL for RV
- Problems
- 4 Maxims/Requirements
- LTL Semantics
  - 2-valued: FLTL (1)
  - 3-valued: LTL3 (2)
  - 4-valued: RV-LTL (3)
- Conclusion
(1) FLTL (2-Valued)

- LTL with additional weak X operator and slightly different semantics for until/release operator
- Satisfies maxim 1,2
- Weak X operator
- Violates maxim 3,4
- 2-valued
(2) LTL3 (3-valued)

- 3-Valued semantics
  - true, false and inconclusive or short ?
- Idea:
  - Good prefix → true
  - Bad prefix → false
  - Otherwise → '?'
- Satisfies maxim 1, 3, 4
- Violates maxim 2
- No weak X operator
(2) LTL3 (3-valued)

- Every uncertain situation is collapsed to one value: '?'
- Are there different levels of uncertainty?
(3) RV-LTL (4-valued)

- Combine LTL3 and FLTL to satisfy all 4 maxims
- Split inconclusive verdict
Req/Ack schema is always '?' for LTL3 and false for FLTL

Two cases

1st: all req. have been acked up to now
   - But all future req. must also be acked
2nd: at least one req. has still to be acked
   - All future req. must be acked + the still pending
(3) RV-LTL (4-valued)

- 4 truth values
  - true
  - false
  - 1st: presumably true: its likely to be satisfied for all future request
  - 2nd: presumably false: its likely to remain unsatisfied
- RV-LTL satisfies maxims 1,3,4
RV-LTL satisfies maxim 2

1st: complementary truth values in the 4-valued domain are always different

2nd: show that the complement leads to complementary and different truth values

\[ [\omega = \varphi]_{RV} = T \implies [\omega = \varphi]_3 = T \implies [\omega = \neg \varphi]_3 = F \implies [\omega = \neg \varphi]_{RV} = F \]

\[ [\omega = \varphi]_{RV} = [\omega = \varphi]_F = [\omega = \neg \varphi]_F = [\omega = \neg \varphi]_{RV} \]
RV-LTL satisfies maxim 2

1st: complementary truth values in the 4-valued domain are always different

2nd: show that the complement leads to complementary and different truth values

\[ [\omega \models \varphi]_{RV} = T \iff [\omega \models \varphi]_3 = T \iff [\omega \models \neg \varphi]_3 = F \iff [\omega \models \neg \varphi]_{RV} = F \]

\[ [\omega \models \varphi]_{RV} = [\omega \models \varphi]_F = [\omega \models \neg \varphi]_F = [\omega \models \neg \varphi]_{RV} \]
RV-LTL satisfies maxim 2

1st: complementary truth values in the 4-valued domain are always different

2nd: show that the complement leads to complementary and different truth values

\[
[\omega|=\varphi]_{RV} = T \quad [\omega|=\varphi]_{3} = T \quad [\omega|=\neg \varphi]_{3} = F \quad [\omega|=\neg \varphi]_{RV} = F
\]

\[
[\omega|=\varphi]_{RV} = [\omega|=\varphi]_{F} = [\omega|=\neg \varphi]_{F} = [\omega|=\neg \varphi]_{RV}
\]
Conclusion

- 2-valued semantics usually not enough for RV and finite traces
- 3-valued better, but still some drawbacks e.g. regarding liveness properties
- 4-valued RV-LTL fits the “intuitive” semantics for inconclusive verdicts
- Safety and liveness properties can be monitored

Remark:
- Monitor construction is possible in an efficient way
Outlook

- RV can be used for different purposes, not only for pure verification
- Self-healing or self adapting systems:
  - Monitoring the system and detect violations online
  - Change the behavior to meet the requirements again