Exam solution – Initiation à la vérification

January 12, 2016

1. Petri nets

(a) i. not live, not bounded, not cyclic

\[
\begin{array}{c}
\text{□} \\
\text{□}
\end{array}
\]

ii. not live, not bounded, cyclic: impossible
Suppose that a net (with place \(p\)) is cyclic and unbounded but not live, i.e. there exists a transition \(t\) and a marking \(m \in R\) such that \(m\) cannot reach any \(m'\) with \(m'(p) \geq W(p, t)\).
But due to cyclicity, one can reach \(m_0\) from any \(m \in R\), and due to unboundedness, \(R = \text{reach}(m_0)\) contains a marking \(m'\) with \(m'(p) \geq W(p, t)\), a contradiction.

iii. not live, bounded, not cyclic

\[
\begin{array}{c}
\text{□} \rightarrow \text{□}
\end{array}
\]

iv. not live, bounded, cyclic

\[
\begin{array}{c}
\text{□} \\
\text{□}
\end{array}
\]

v. live, not bounded, not cyclic

\[
\begin{array}{c}
\text{□} \\
\text{□}
\end{array}
\]

vi. live, not bounded, cyclic

\[
\begin{array}{c}
\text{□} \\
\text{□}
\end{array}
\]

vii. live, bounded, not cyclic: impossible
Let \(p\) be the single place. Call a transition \(t\) increasing if \(W(p, t) < W(t, p)\), preserving if \(W(p, t) = W(t, p)\), and decreasing if \(W(p, t) > W(t, p)\).
Suppose that the net is bounded but not cyclic. Boundedness implies that no transition can be increasing. If the net had preserving transitions only, then only the initial marking is reachable, and the net would be cyclic. Thus the net must have at least one decreasing transition \(t\). Consider the run where we repeat \(t\) until the number of tokens is less than \(W(p, t)\). After this, \(t\) can never fire again, hence the net is not live.

viii. live, bounded, cyclic

\[
\begin{array}{c}
\text{□} \\
\text{□}
\end{array}
\]

(b) Let \(I\) be a positive invariant, \(m\) some reachable marking and \(q\) some place. We have

\[
m(q) \leq m(q) \cdot I(q) \leq \sum_{p \in P} I(p) \cdot m(p) = \sum_{p \in P} I(p) \cdot m_0(p).
\]
The first two steps are justified by the fact that I is positive and \( m(p) \geq 0 \) for all \( p \). The last step follows from the fact that I is an invariant. The latter expression is a constant and provides a bound for \( q \) (and in fact for all places).

The statement is false. The net shown below is live and can reach any odd number of tokens. However, if \( m(p) = 2 \), the net can reach 0 tokens and is unable to continue afterwards.

\[
\begin{array}{c}
\square & \rightarrow & \bullet & \rightarrow & \square
\end{array}
\]

2. BDDs and Abstraction

For a set of variables \( X \), let \( 2^X \) denote the set of Boolean assignments over \( X \). For an assignment \( A \), \( F[X/A] \) denotes the formula where occurrences of all \( x \in X \) in \( F \) are replaced by \( A(x) \). Models of formulae are taken to be assignments over \( V \), and variables from \( V \) do not occur in \( Y \).

\( \exists X \) – Let \( A \) be any assignment to \( x \). The last step follows from the fact that \( X \) is a constant and provides a bound for \( q \) (and in fact for all places).

\[
\begin{array}{c}
\text{if } F = 0 \text{ or } F = 1 & F \\
\text{if } G = 1 & G \\
\text{if } F = G & F \\
\text{if } \text{top}(F) < \text{top}(G) & \text{inter}(F \lor F_1, G) \\
\text{if } \text{top}(F) > \text{top}(G) & \text{inter}(F, G_0 \land G_1) \\
\text{if } x = \text{top}(F) = \text{top}(G) & \text{mk}(x, \text{inter}(F_1, G_1), \text{inter}(F_0, G_0))
\end{array}
\]

\( \text{mk}(x, \text{inter}(F_1, G_1), \text{inter}(F_0, G_0)) \) if \( x = \text{top}(F) = \text{top}(G) \)
(d) The drawing below gives the BDDs for $F$ (root node $a$) and $G$ (root node $d$).

The equation from (c) yields:

\[ \text{inter}(a, d) = \text{inter}(b \lor 0, d) = \text{inter}(b, d) = \text{mk}(x_2, \text{inter}(0, e), \text{inter}(c, 1)) = \text{mk}(x_2, 0, 1). \]

The resulting BDD represents the formula $\neg x_2$, which is indeed implied by $F$ and implies $G$.

3. Partial-order reduction

(a) The following states have multiple enabled actions:
- $s_2$ with \{a, c, e\}: of the three pairs, only $\langle a, e \rangle$ form a ‘diamond’.
- $t_2$ with \{b, c, e\}: dito for $\langle b, c \rangle$.
- $t_1$ with \{b, d\}: we conclude that $\langle b, d \rangle$ are not independent.
- $s_3$ with \{a, f\}: dito for $\langle a, f \rangle$.

Thus, the only relevant independent pairs are $\langle a, e \rangle$ and $\langle b, c \rangle$.

Obviously, only $d$ and $f$ are visible, the other actions are invisible.

(b) Not a single transition can be removed according to rules C0–C3. In fact, it suffices to apply rules C0 and C1, due to the dependencies found in (a).

(c) There are three classes for stutter equivalence to preserve: a run (i) either remains in the white states, (ii) or eventually reaches the black states, (iii) or eventually reaches the grey states. The only loop in the white states is between $s_2$ and $t_2$, so these two must be kept for (i). To preserve (ii), we can eliminate either $s_1$ or $t_1$ with their adjacent transitions. For (iii), the analogue holds with $s_3$ and $t_3$. A possible result is shown below; in any case six transitions are eliminated.