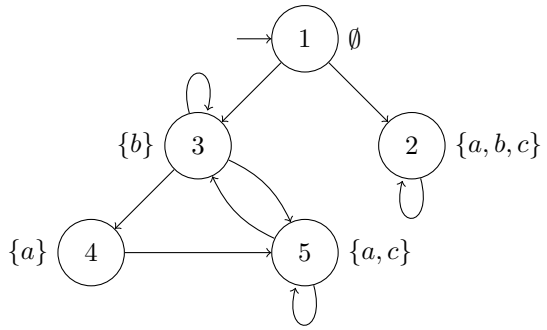
Software verification - Linear Temporal Logic and transition systems

assignments for week 1

Assignment 3 should be handed in in the post box at floor 1 of Mercator 1, behind the couches, before monday 17-04-2017, 12:30.

1 Model checking by hand

Consider the following transition system:



For each of the LTL-formulas below, check whether the formula holds for this transition system. If so, argue why. If not, give a counter example. Formally, a counter-example to an LTL-formula is an infinite sequence of sets of atomic propositions, but with a transition system, a sequence of states is clearer.

a) $\Box\Diamond a$

b) $\Diamond\Box a$

c) $b \vee \bigcirc\bigcirc b$

d) $\bigcirc(a \mathbf{U} b)$

e) $\bigcirc(b \mathbf{U} a)$

f) $\Box\Diamond(\neg b \rightarrow c)$

g) $\Box(b \vee c \vee \bigcirc(b \vee c))$

h) $(\bigcirc\bigcirc c) \mathbf{U} b$

2 Equivalence proofs

Prove that the following equivalences hold, or give counterexamples why they do not hold:

a) $\Diamond\Box\phi = \Box\Diamond\phi$

b) $\bigcirc\Diamond\phi = \Diamond\bigcirc\phi$

c) $\Diamond\phi \vee \Diamond\psi = \Diamond(\phi \vee \psi)$

d) $\Diamond\phi \wedge \Diamond\psi = \Diamond(\phi \wedge \psi)$

3 Expressing the ABP in LTL

Consider two network devices, A and B, following the alternating bit-protocol to send messages from A to B:

- If A sends a message, it contains a sequence-bit. It then keeps resending the message until it receives the same bit from B as acknowledgement. Only then may A send the next message, with a different sequence bit.
- B simply receives messages, and sends back the sequence bit as acknowledgement. It keeps resending it, until it receives a new sequence bit.

The atomic propositions A_{S0} and A_{R0} denote that A sends or receives bit 0, respectively. A_{S1}, A_{R1}, B_{S0}, B_{R0}, B_{S1} and B_{R1} are defined similarly (the contents of messages are not considered).

Formalize the following properties in LTL, or explain why it cannot be expressed in LTL:

a) An obvious constraint: A can only send one thing at a time.

b) A is continuously sending, without pausing.

c) A will never stop sending, but he may pause for a while.

d) Whenever A sends a bit, it will be acknowledged by B.

e) Whenever A sends a bit, B will receive it immediately.

f) Whenever B receives a 0, it will keep acknowledging 0 as long as it does not receive a 1.

g) Whenever A sends a 0, it will not send a 1 before receiving the acknowledgement for 0.

h) A can always send.

i) The network is lossy: if A sends, B may or may not receive.