LUDWIG-
MAXIMILIANS-
UNIVERSITÄT
MÜNCHEN

Übung 10 – Temporale Logik

Formale Techniken in der Software-Entwicklung

Christian Kroiß

Quellen

Die Beispiele auf den Folien 3-6 stammen aus dem folgenden Buch:

Michael Huth and Mark Ryan

"Logic in Computer Science"

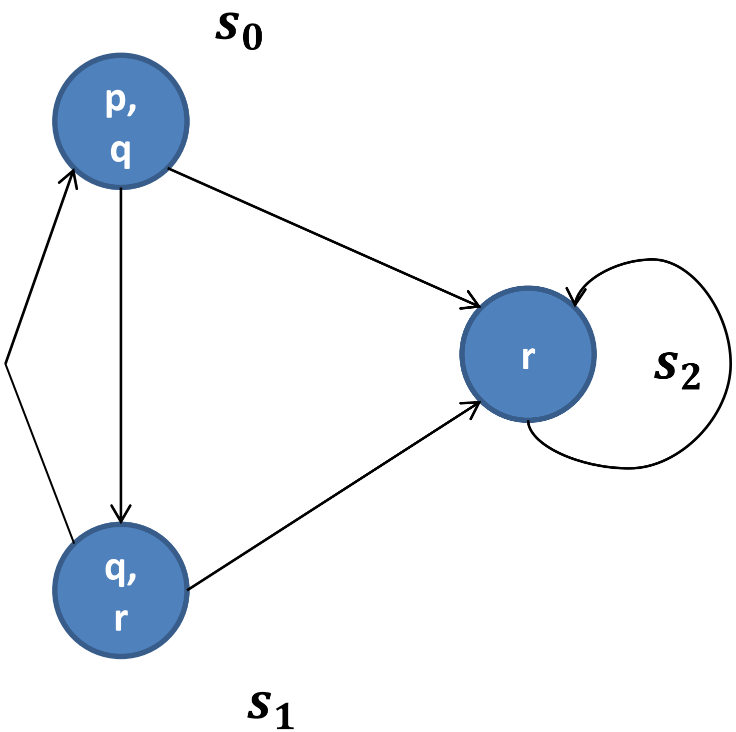
2nd edition

ISBN 978-0-521-54310-1

Die Fragen von den Folien 7,9 und 11 stammen von der dazugehörigen Webseite: http://www.cs.bham.ac.uk/research/projects/lics/tutor/chap3/questions.html

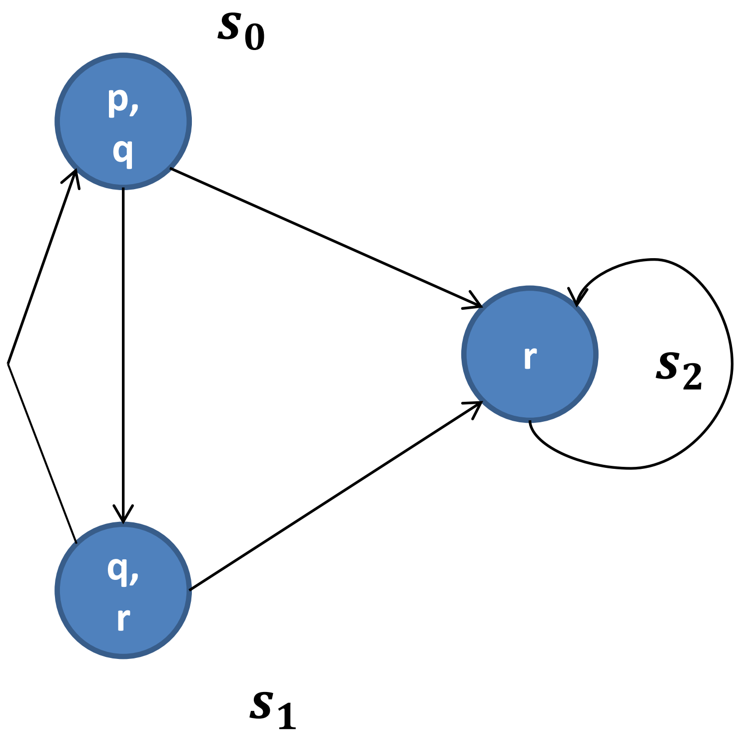
Dort findet man auch detaillierte Erklärungen zu den richtigen und falschen Antworten, wenn man auf sie klickt.

LTL



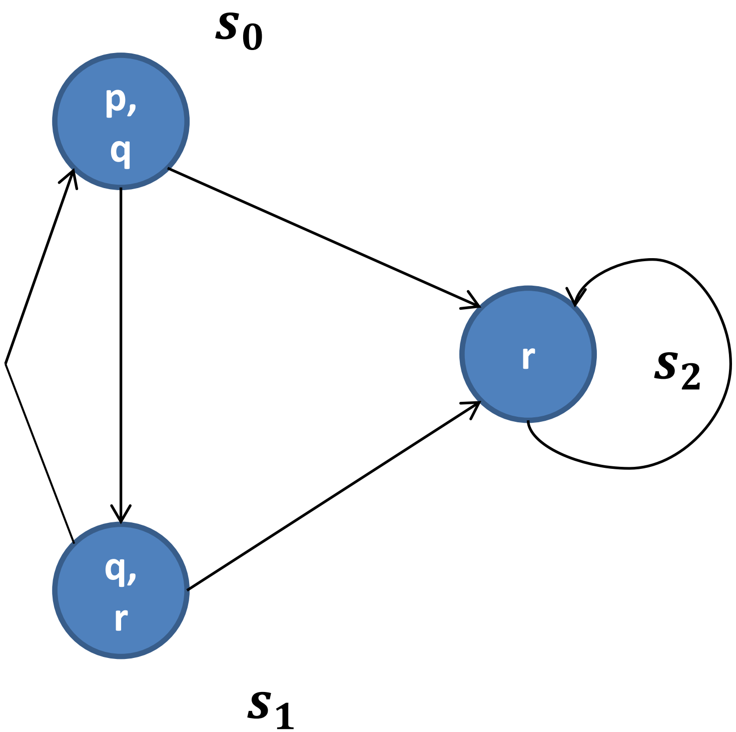
1. $M, s_0 \vDash p \wedge q$
2. $M, s_0 \vDash \top$
3. $M, s_0 \vDash \neg r$
4. $M, s_0 \vDash Xr$
5. $M, s_0 \nvDash X(q \wedge r)$
6. $M, s_0 \vDash G\neg(p \wedge r)$
7. $M, s_2 \vDash G\, r$
8. For any s:
   $M, s \vDash F(\neg q \wedge r) \rightarrow \mathrm{FG\ r}$

LTL



9. $GF\ p$
is satisfied by
$s_0 \to s_1 \to s_0 \to s_1 \to$ ...
but not by
$s_0 \to s_2 \to s_2 \to \cdots$

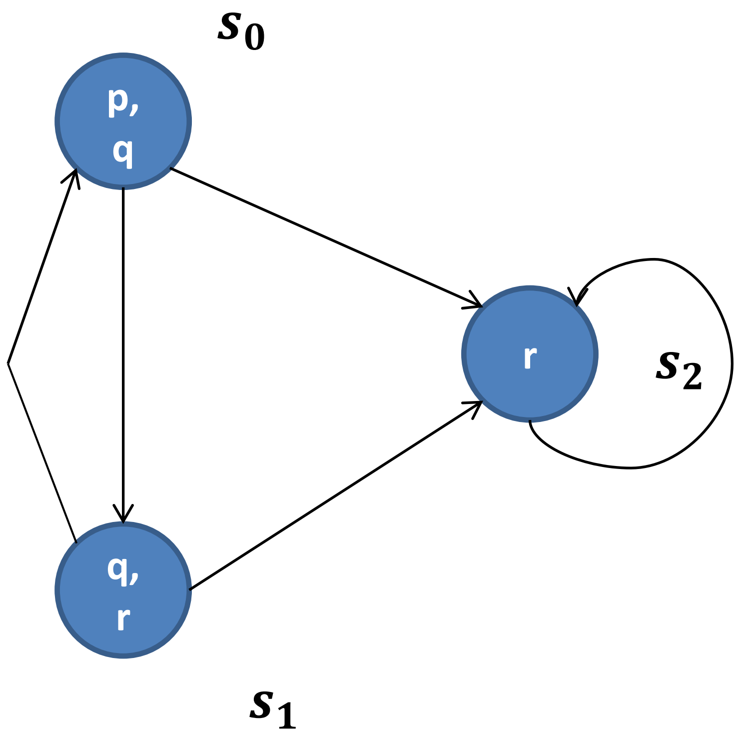
10. $M, s_0 \vDash GF\ p \ \to GF\ r$
but: $M, s_0 \nvDash GF\ r \ \to GF$ p



CTL

1. $M, s_0 \vDash p \wedge q$
2. $M, s_0 \vDash \neg r$
3. $M, s_0 \vDash EX\ (q \wedge r)$
4. $M, s_0 \vDash \neg AX\ (q \wedge r)$
5. $M, s_0 \vDash \neg EF (p \wedge r)$
6. $M, s_2 \vDash EG\ r$
7. $M, s_0 \vDash AF\ r$
8. $M, s_0 \vDash E[(p \wedge q)\ U\ r]$

CTL



9. $M, s_0 \models A[p\ U\ r]$
10. $M, s_0 \models AG(p \vee q\ Foto)$
11. $M, s_0 \models AG(p \vee q \vee r \rightarrow EF\ EG\ r)$

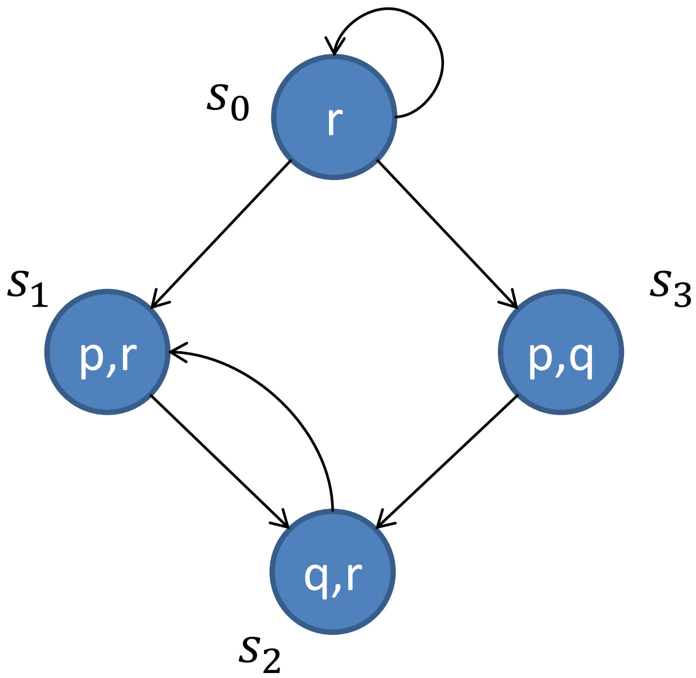
Which of the specifications in plain English below convey the mathematical meaning of the CTL formula
$AG(p \rightarrow A[qUr])$?

1. Any reachable state in which p is true has a path from it on which r is eventually true, and until then q is true.
2. If p is true in every reachable state, then there is a path along which q is continuously true, until r becomes true.
3. If p is true in every reachable state, then for any path along which q is continuously true, r becomes true.
4. For any reachable state in which p is true, then, on any path from that state, q is continuously true until r becomes true, and r is guaranteed to become true.
5. If p is true in every reachable state, then on every path there is a state at which r is true, and q is true continuously until then.

Which of the specifications in plain English below convey the mathematical meaning of the CTL formula

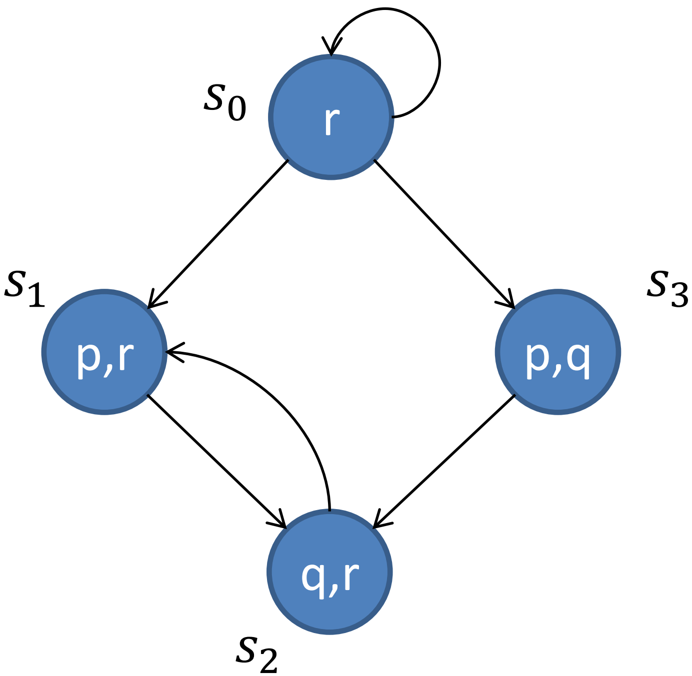
$AG(p \rightarrow A[qUr])$?

1. Any reachable state in which p is true has a path from it on which r is eventually true, and until then q is true.
2. If p is true in every reachable state, then there is a path along which q is continuously true, until r becomes true.
3. If p is true in every reachable state, then for any path along which q is continuously true, r becomes true.
4. For any reachable state in which p is true, then, on any path from that state, q is continuously true until r becomes true, and r is guaranteed to become true.
5. If p is true in every reachable state, then on every path there is a state at which r is true, and q is true continuously until then.



Which of the CTL formulas below are satisfied in state s_0?

1. $AF(q \wedge r)$
2. $AG\big(p \rightarrow AF(p \wedge r)\big)$
3. $A[rUq]$
4. $AG\big(p \rightarrow AG(p \vee q)\big)$
5. $AG\ EF\ \neg r$



Which of the CTL formulas below are satisfied in state s_0?

1. $AF(q \wedge r)$
2. $AG(p \rightarrow AF(p \wedge r))$
3. $A[rUq]$
4. $AG(p \rightarrow AG(p \vee q))$
5. $AG\ EF\ \neg r$

Which of the following pairs of CTL formulas are equivalent ?

1. EFp and EGp
2. $EFp \vee EF\ q$ and $EF(p \vee q)$
3. $AFp \vee AFq$ and $AF(p \vee q)$
4. AFp and $A[p\ U\ \top]$
5. $EF\ \neg p$ and $\neg AFp$

Which of the following pairs of CTL formulas are equivalent ?

1. EFp and EGp
2. $EFp \vee EF\ q$ and $EF(p \vee q)$
3. $AFp \vee AFq$ and $AF(p \vee q)$
4. AFp and $A[p\ U\ \top]$
5. $EF\ \neg p$ and $\neg AFp$