



Model Checking

H. Peter Gumm

Philipps-Universität Marburg

Sommersemester 2007



Inhaltsverzeichnis

1. Einführung

- Motivation
- Ampel-Beispiel
- Modellierung in SMV
- Temporale Eigenschaften

2. Das SMV System

- SMV Module und Prozesse
- Schaltungen, Controller,
- Protokolle (Mutex, AltBit)
- Verifikation temporaler Eigenschaften

3. Kripke-Strukturen und SMV

- Kripke-Strukturen
- Programme
- Verteilte Programme
- Modellierung in SMV

4. Lineare Temporale Logik

- Berechnungen und Spuren
- LTL Syntax und Semantik
- Spezifikationsmuster
- Lineare Äquivalenz

5. Verzweigungslogik

- CTL Syntax und Semantik
- Ausdrucksstärke, Trennschärfe
- Vergleich LTL, CTL, CTL*
- Berechnungsbäume

6. CTL Model Checking ?

- Reduktion
- EX, AX, EU, EG
- Fixpunkte und ihre Berechnung
- Fairness

7. LTL Model Checking

- Büchi-Automaten
- LTL Model Checking
- LTL-Formeln und Büchi-Automaten
- Tableau-Konstruktion

8. Symbolisches Model Ckecking

- Repräsentation von Daten
- Zustandsexplosion
- Kripke-Strukturen als Boolesche Terme
- Datenstruktur: OBDD

9. Bounded Model Ckecking

- Gegenbeispiele
- Endliche Pfade
- BMC als SAT-Problem
- SAT-Prüfer

10. Abstrakte Temporale Eigenschaften

- Safety
- Liveness
- Satz v. Alpern Schneider
- Der metrische Raum S^ω



Literatur



Wichtig

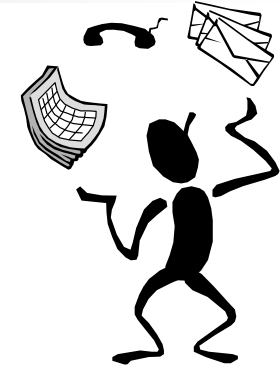
- Michael Huth, Mark Ryan: *Logic in Computer Science: Modelling and Reasoning about Systems*. Cambridge University Press, Second Edition, 2004.
Kapitel 3 und 6 im Netz.
- Joost-Pieter Katoen: *Concepts, Algorithms and Tools for Model Checking*. Vorlesungsskript Uni-Erlangen, 1999.
<http://www.cs.aau.dk/~kgl/TECS06/katoen2.ps>
- Kenneth McMillan: *Symbolic Model Checking*. Doktorarbeit, Carnegie Mellon University, 1992.
<http://www.kenmcmil.com/>

Weiterhin

- Karsten Schmidt: *Model Checking*. (HTML-Skript)
<http://www.informatik.hu-berlin.de/~kschmidt/modelchecking/>
- Henrik Reif Andersen: *An Introduction to Binary Decision Diagrams*.
<http://www.it.edu/courses/EAP/Notes/bdd-eap.pdf>
- E.M. Clarke, O. Grumberg, and D. Peled. *Model Checking*. MIT Press, 1999.
Im Semesterapparat.



Organisatorisches



■ Übungen

- ☐ Tragen Sie sich in die Übungslisten ein
- ☐ Regelmäßige Teilnahme
- ☐ Aktive Mitarbeit
- ☐ Individuell gelöste Hausaufgaben
- ☐ Programmieraufgaben
- ☐ Abgabe elektronisch
- ☐ Verspätet abgegebene Lösungen werden nicht gewertet

■ Verantwortlich

- ☐ Dipl. Math. Egbert Fohry

fohry@mathematik.uni-marburg.de

■ Vorlesungsseite:

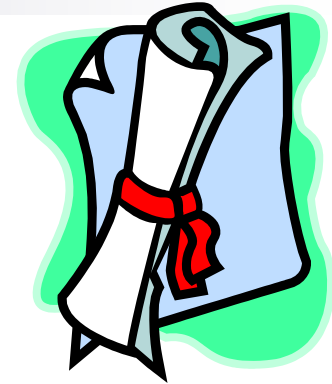
- ☐ www.mathematik.uni-marburg.de/~fohry

- ☐ Besuchen Sie die Vorlesungsseite regelmäßig:

- Ankündigungen
- Korrekturen, Hinweise
- Musterlösungen
- Terminänderungen
- Ergebnisse, etc.



Scheinkriterien



■ Unbenoteter Schein:

- ☐ Regelmäßige Teilnahme an den Übungen
- ☐ Aktive Mitarbeit
- ☐ 50% der möglichen Übungspunkte
- ☐ Max. 2 Scheine unbearbeitet

■ Benoteter Schein:

- ☐ Unbenoteter Schein
+ bestandene Abschlussklausur

falls nicht genügend Teilnehmer:
Klein-Projekt + Kolloquium



Software

■ Software

☐ NuSMV

- <http://nusmv.irst.itc.it/>

☐ Bequem dazu: UltraEdit

- Uni Marburg besitzt genügend Lizenzen

☐ oder analoge Umgebung

- z.B. Emacs, ...





Installation



- Herunterladen

- Nusmv 2.4.1 von

- <http://nusmv.first.itc.it>

- Installation

- Nusmv:

- setup aufrufen

- Pfad setzen zu

- C:\Program Files\NuSMV-2.4.1\bin



NuSMV



□ Ausprobieren:

- Aus einer DOS-Shell **NuSMV.exe** aufrufen
- Aus einer DOS-Shell NuSMV mit einer smv-Datei (z.B. "**Ampel.smv**") aufrufen:

C:> **NuSMV Ampel.smv**

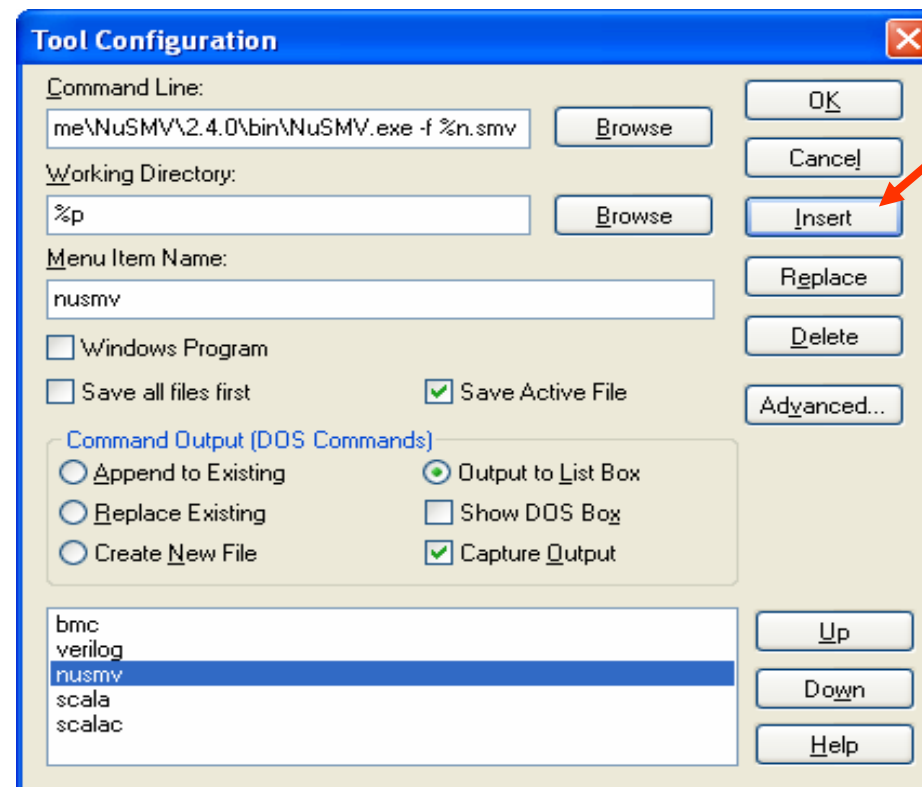
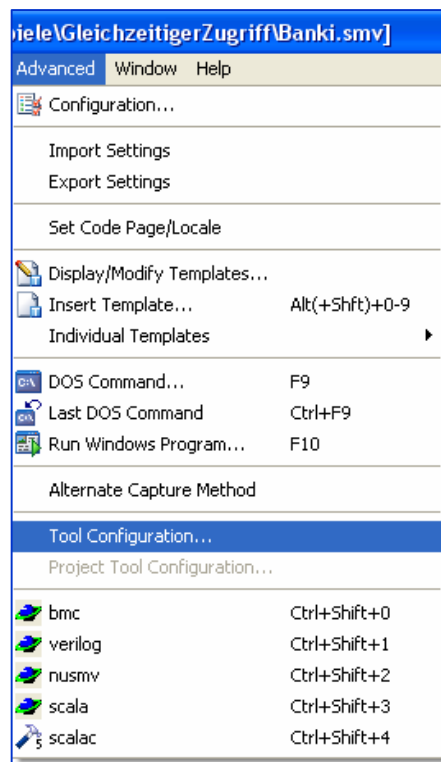
```
C:\cygwin\usr\local\nusmv\bin\NuSMV.exe
*** This is NuSMV 2.1.2 (compiled 2002-11-22 12:00:00)
*** For more information of NuSMV see <http://nusmv.first.itc.it>
*** or email to <nusmv-users@first.itc.it>.
*** Please report bugs to <nusmv-users@first.itc.it>.
```




Anpassung an UltraEdit*



- *Advanced -> Tool Configuration*
- Command Line: `C:\Programme\NuSMV\2.4.0\bin\NuSMV.exe -f %n.smv`
- Working Directory: %p
- Menu Item Name: **NuSMV**
- *Output to Listbox*, *SaveActiveFile* und *Capture Output* markieren
- Auf *Insert* klicken.





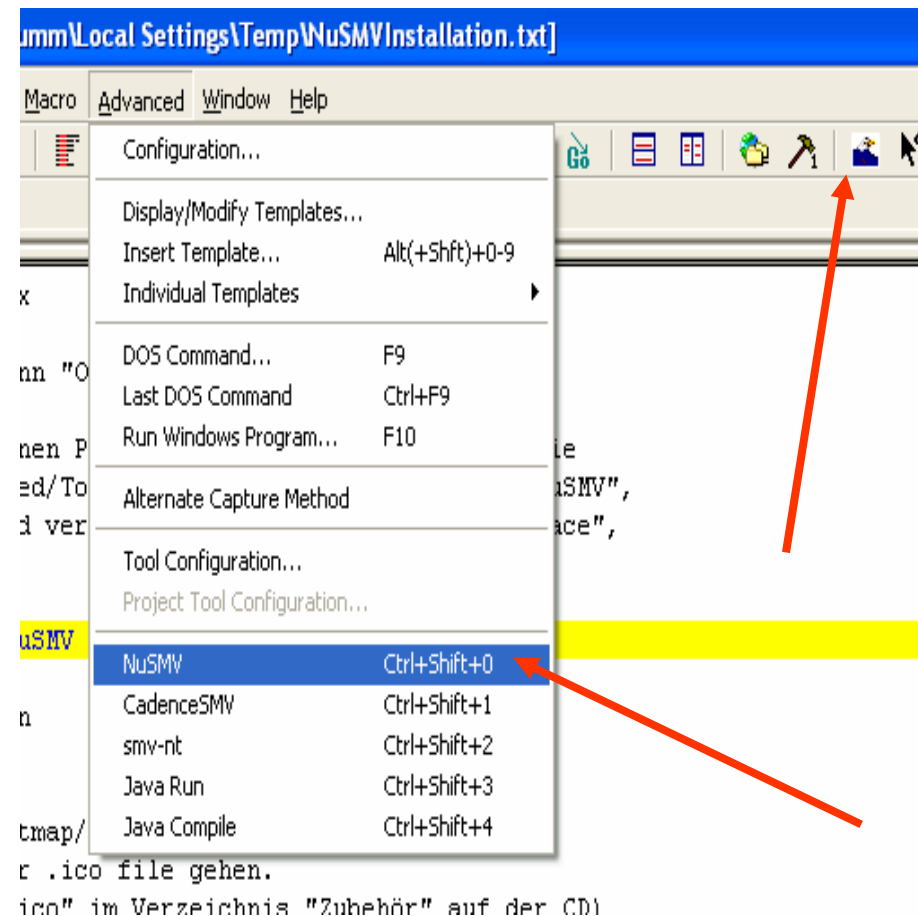
Syntax Highlighting für Ultraedit

- NuSMV-Wordlist für Ultraedit herunterladen von
 - `www.dii.unisi.it/~hybrid/tools/mex/downloads/nuSMV-wordfile.txt`
- Text in Datei
 - `wordfile.txt` im Ultraedit-Verzeichnis kopieren
 - Erste Zeile in wordlist beginnt mit
 - `/L13"nuSMV" Line Comment = -- ...`
 - `/Lxx` bezeichnet die `xx`-te Sprache (language), die in `wordfile.txt` definiert ist
 - Diese Nummerierungen müssen fortlaufend sein
 - `/Lxx` ggf. anpassen, z.B. `/L10`



NuSMV aus UltraEdit aufrufen

- .smv-Datei in Ultraedit laden
- NuSMV aufrufen
 - Drei Möglichkeiten:
 - **Advanced -> NuSMV**
 - **Ctrl+Shift+0**
 - Icon anklicken
- Bei smv-Syntaxfehler
 - Klicken auf die Zeile mit der Fehlermeldung
 - Cursor springt an die richtige Zeile.





Beispiel

- Inputfile: Ampel5.smv



- NuSMV-Testlauf mit Ergebnis

```
UltraEdit-32 - [G:\SS03VLModelChecking\SMV-Beispiele\Ampel\ampel5.smv]
File Edit Search Project View Format Column Macro Advanced Window Help
Installation von NuSMV unter Windows.txt wordfile.txt ampel5.smv

-- Fuenfte Version einer Ampelschaltung
-- Modularisierung

-- Wir erklären einen MODUL zaehler -----
--
MODULE zaehler(ampel)
VAR
    counter : 0 .. 5;
ASSIGN
    init(counter) := 0;
    next(counter) := case
        !ampel : 0 ;
        1: (counter+1) mod 6;
    esac;

-----
-- Ist ein Zaehler als Instanziierung des Moduls erklärt
-- etwa als
--     VAR z : zaehler(E_Ampel);
-- So kann auf die Variable counter zugegriffen werden durch
--     z.counter
-----

MODULE main
VAR
    E_Sensor : boolean;
    S_Sensor : boolean;
    N_Sensor : boolean;

*** Please report bugs to <nusmv@irst.itc.it>.
-- specification G (E_Ampel -> (!N_Ampel & !S_Ampel)) is true
-- specification G (E_Sensor -> F E_Ampel) is true
-- specification G (S_Sensor -> F S_Ampel) is true
-- specification G (N_Sensor -> F N_Ampel) is true
-- specification G F (!N_Sensor & N_Ampel) is true
-- specification G F (!S_Sensor & S_Ampel) is true
```