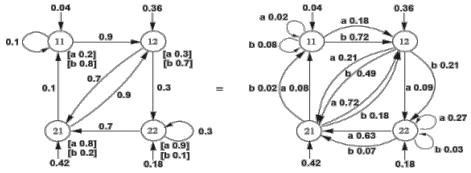
Recent Advances in Model Checking (IN0012, IN2106)

Practical course

Prof. J. Kretinsky, J. Eisentraut, A. Evangelidis, S. Mohr, M. Weininger

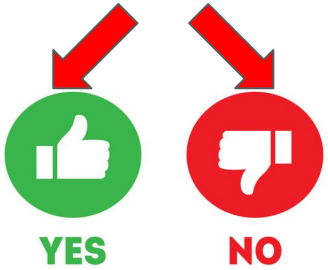
Context



$\mathsf{A}(\psi \vee \varphi)$
$\mathcal{K} \models \mathsf{A}\neg(\neg\psi \wedge \neg\varphi)$
$\mathcal{K} \models \mathsf{A}\neg(\neg\langle\mathcal{A}_\exists(\{\}, 1, 1, \neg\psi) \wedge \mathcal{A}_\exists(Q, \varphi_\mathcal{I}, \varphi_\mathcal{R}, \varphi_\mathcal{F})))$
$\mathcal{K} \models \mathsf{A}\neg\mathcal{A}_\exists(Q, \varphi_\mathcal{I}, \varphi_\mathcal{R}, \neg\psi \wedge \varphi_\mathcal{F})$
$\mathcal{K} \models \mathsf{A}\mathcal{A}_\forall(Q, \varphi_\mathcal{I}, \varphi_\mathcal{R}, \psi \vee \neg\varphi_\mathcal{F})$
$\mathcal{K}_\times \models \varphi_\mathcal{I} \rightarrow \mathsf{A}(\psi \vee \neg\varphi_\mathcal{F})$
$\mathcal{K}_\times \models p \rightarrow \mathsf{A}(\neg a \vee \neg\mathsf{GF}(p \wedge b))$
$\mathcal{K}_\times \models p \rightarrow \mathsf{A}(\neg a \vee \mathsf{FG}\neg(p \wedge b))$
$\mathcal{K}_\times \models p \rightarrow (\neg a \vee \mathsf{AFG}\neg(p \wedge b))$
$[\![\mathsf{AFG}\neg(p \wedge b)]\!]_{\mathcal{K}_\times} = \mathcal{S} \setminus [\![\mathsf{EGF}(p \wedge b)]\!]_{\mathcal{K}_\times} = \{\}$
$\mathcal{K}_\times \models p \rightarrow \neg a$
$[\![(p \rightarrow \neg a)]\!]_{\mathcal{K}_\times} = \{SQ0, SQ1\}$

- Quantitative (e.g. probabilistic), more agents, several competing properties,...
- Well-established industrial method & recent research

Model Checker



Content

1. Understand the ideas of a recent scientific publication.
2. Implement them.
3. Test them.

Different focus, depending on paper:

Theory: Understanding and extracting the ideas	Implementation: Technically involved	Evaluation: Comparing multiple ideas

Publications (and hence focus) will be selected after a short introductory lecture phase.

Structure

~4 weeks: Introductory lectures about
theory common to all papers and relevant software to build on

~4 weeks: Understanding paper, developing prototype
Groups of up to 3 people, mostly independent.

10%

Midterm presentation: Convince us that you are on the right track

50%

30%

~4 weeks: Finishing implementation, writing documentation

10%

Endterm presentation: Demonstrate what you achieved

What do we expect?

- Working code that we can execute on several examples and reproduce your results.
- Documentation that allows us to find and understand the most important methods of your code.
- Endterm presentation to demonstrate that you solved the problem.
- Midterm presentation to demonstrate that you are on the right track.
  E.g. by showing understanding and identifying missing parts;
  unit tests for existing code; and dummy methods for missing parts.

More details follow in the actual course.

Which questions remain unanswered?

Addendum: To be preferred in the matching, send a mail to maxi(dot)weininger(at)tum(dot)de, giving your name and matriculation number.