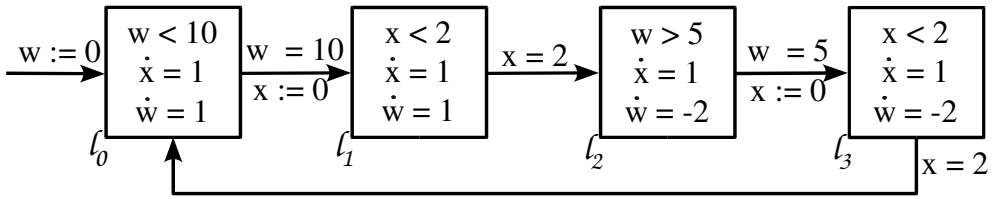
Abstract interpretation of temporal logic: abstract model checking revisited

John Gallagher[1,2] Gourinath Banda[1] Pierre Ganty[2]

[1]Roskilde University [2]IMDEA Software, Madrid

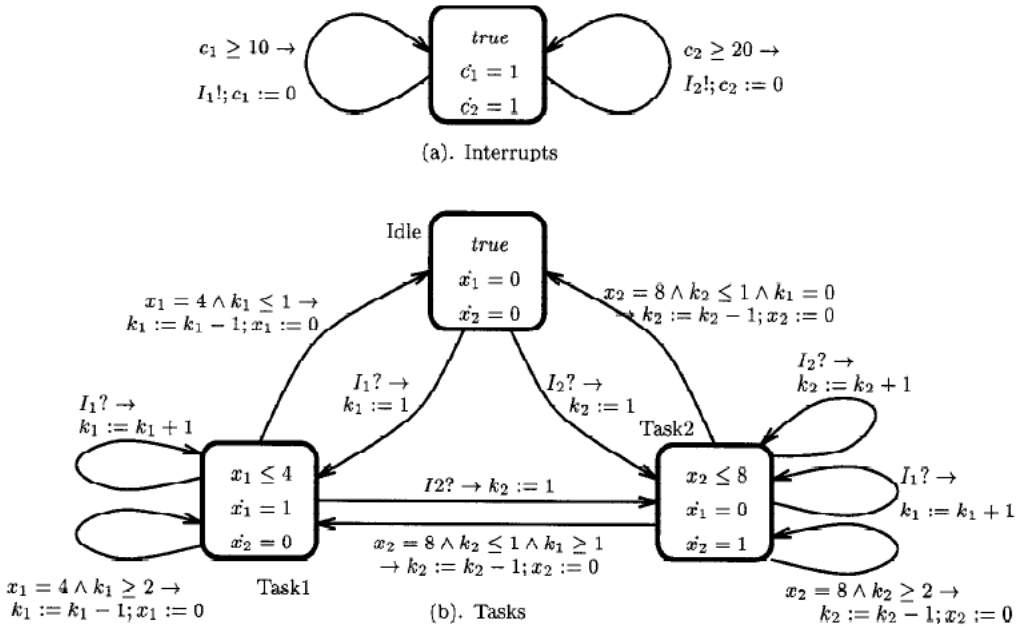
DANSAS 2010, Odense

Example 1: Temporal properties of a water level controller [Halbwachs et al. 94]



- Safety property: $AG(0 \leq w \wedge w \leq 12)$. ($w$ always stays between 0 and 12)
- Existential properties. $EF(w = 10)$. (w can reach the value 10).
- Eventual safety (nested CTL property). $AF(AG(1 \leq w \wedge w \leq 12))$. (Eventually, w remains between 1 and 12).

Example 2: Properties of a task scheduler [Halbwachs et al. 94]



- $AG(k2 > 0 \rightarrow AF(k2 = 0))$. (A waiting high priority task is eventually scheduled).
- $EF(k2 = 1)$. (A high priority task can arise).
- $AG(k2 \leq 1)$. (No more than one high priority task can be waiting).

The model checking framework

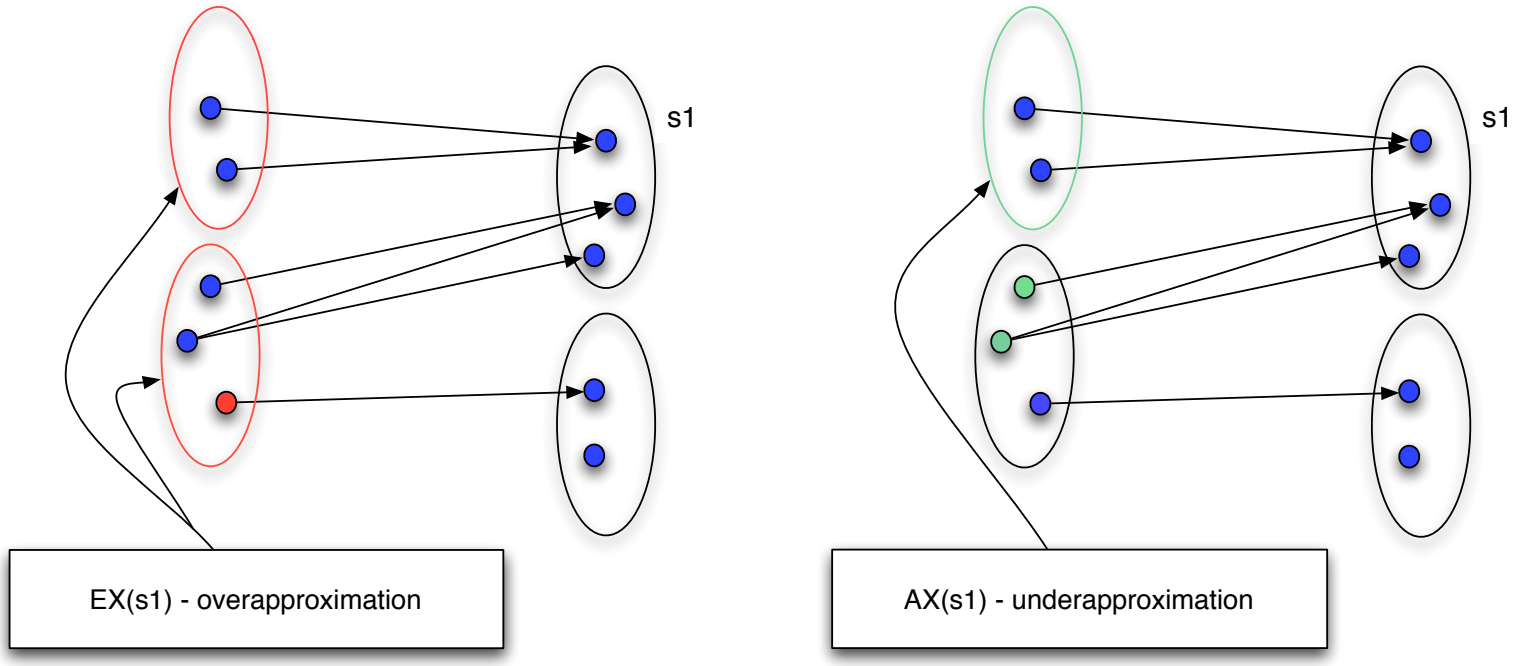
- Express system as a transition system (Kripke structure).
- Evaluate temporal properties over the structure.
- Given a property ϕ, $[\![\phi]\!]$ denotes the states at which ϕ holds.
- Model checking algorithms are essentially algorithms for evaluating $[\![\phi]\!]$. The algorithms terminate if the structure is finite.

Model checking infinite-state transition systems (Kripke models)

- The original approach [Clarke, Grumberg & Long, 1992].
- Define a finite partition of the infinite set of states.
- Partition induces a finite abstract transition system which *over*-approximates the concrete transition relation
- Model checking in such an abstract transition system is sound only for universal properties (or for refutation of existential properties).

The need for Over- and Under-approximations in Abstract Transition Systems

Let s1 be some abstract state (a set of concrete states).



One approach: Dual (Modal/Mixed) Transition Systems

- In order to handle both universal and existential properties, various authors suggest creating two abstract transition systems.
- e.g. [Larsen & Thomsen 1988], [Dams, Gerth & Grumberg 1997], [Godefroid, Huth & Jagadeesan 2001]
- One (may-transitions) for proving existential properties, the other (must-transitions) for universal properties.
- $\Rightarrow$ a modified model checking algorithm and extra correctness proofs

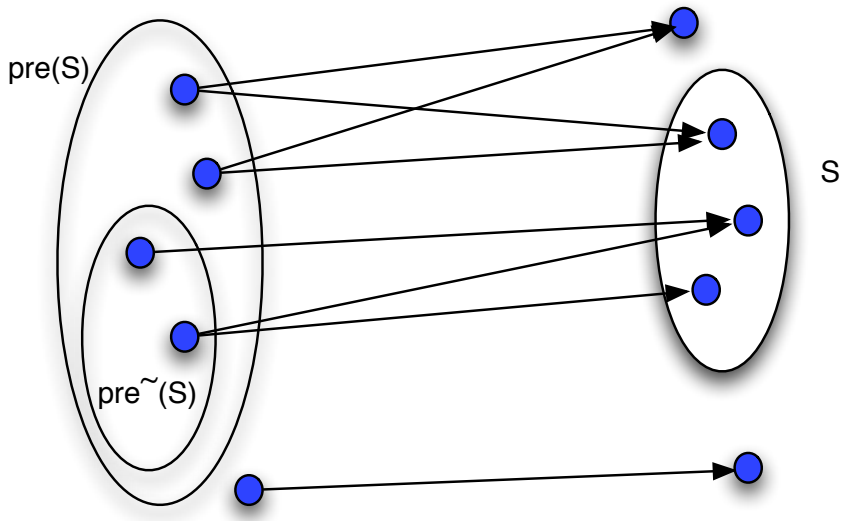
Abstract interpretation approach: Concrete Semantics

Semantics of the μ-calculus (more general than CTL, LTL, etc).

$$
\begin{array}{ll}
[\![Z]\!]_\mu\sigma & = \sigma(Z) \\
[\![p]\!]_\mu\sigma & = \mathit{states}(p) \\
[\![\neg p]\!]_\mu\sigma & = \mathit{states}(\neg p) \\
[\![\phi_1 \vee \phi_2]\!]_\mu\sigma & = [\![\phi_1]\!]_\mu\sigma \cup [\![\phi_2]\!]_\mu\sigma \\
[\![\phi_1 \wedge \phi_2]\!]_\mu\sigma & = [\![\phi_1]\!]_\mu\sigma \cap [\![\phi_2]\!]_\mu\sigma \\
[\![EX\phi]\!]_\mu\sigma & = \mathit{pre}([\![\phi]\!]_\mu\sigma) \\
[\![AX\phi]\!]_\mu\sigma & = \widetilde{\mathit{pre}}([\![\phi]\!]_\mu\sigma) \\
[\![\mu Z.\phi]\!]_\mu\sigma & = \mathrm{lfp}(F) \\
 & \quad \text{where } F(S') = [\![\phi]\!]_\mu\sigma[Z/S'] \\
[\![\nu Z.\phi]\!]_\mu\sigma & = \mathrm{gfp}(F) \\
 & \quad \text{where } F(S') = [\![\phi]\!]_\mu\sigma[Z/S']
\end{array}
$$

pre and $\widetilde{pre}$ functions

A Kripke structure $K = \langle S, \Delta, I, L, \mathcal{P} \rangle$.
Functions *pre* : $2^S \to 2^S$, $\widetilde{pre}$: $2^S \to 2^S$, states : $\mathcal{P} \to 2^S$ as follows.



- states(p) $= \{s \in S \mid p \in L(s)\}$ returns the set of states where $p \in \mathcal{P}$ holds.

Galois connection

- Let S be set of concrete states.
- Let A be set of abstract states.
- $\langle 2^S, \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle 2^A, \subseteq \rangle$ is a Galois connection between the complete lattices $\langle 2^S, \subseteq \rangle$ and $\langle 2^A, \subseteq \rangle$.
- $\alpha : 2^S \to 2^A$ and $\gamma : 2^A \to 2^S$ are adjoint functions.
- $\alpha(X) \subseteq Y \equiv X \subseteq \gamma(Y)$.

Abstract *pre*, $\widetilde{pre}$ and states functions

The functions *pre*, $\widetilde{pre}$ are monotonic.
Given a Galois connection $\langle 2^S, \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle 2^A, \subseteq \rangle$, and a Kripke structure, define

$$\begin{aligned} apre &= \alpha \circ pre \circ \gamma \\ \widetilde{apre} &= \alpha \circ \widetilde{pre} \circ \gamma \\ astates &= \alpha \circ states \end{aligned}$$

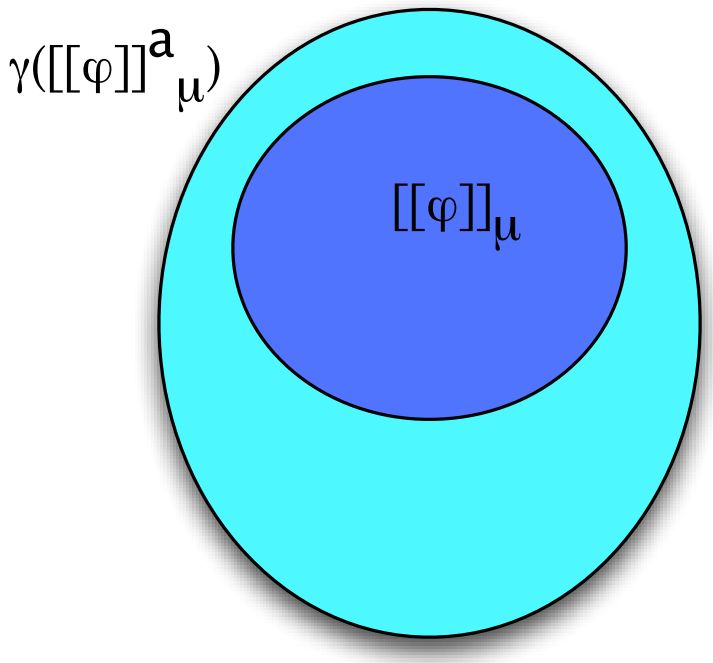
These are optimal for a given Galois connection. We can also take any upper-approximation.

Abstract Semantics of the μ-calculus

The abstract μ-calculus semantic function $[\![\cdot]\!]^a_\mu : \mathrm{Mu} \to (\mathcal{V} \to 2^A) \to 2^A$ is defined as follows.

$$
\begin{aligned}
[\![Z]\!]^a_\mu\sigma &= \sigma(Z) \\
[\![p]\!]^a_\mu\sigma &= \mathrm{astates}(p) \\
[\![\neg p]\!]^a_\mu\sigma &= \mathrm{astates}(\neg p) \\
[\![\phi_1 \vee \phi_2]\!]^a_\mu\sigma &= [\![\phi_1]\!]^a_\mu\sigma \cup [\![\phi_2]\!]^a_\mu\sigma \\
[\![\phi_1 \wedge \phi_2]\!]^a_\mu\sigma &= [\![\phi_1]\!]^a_\mu\sigma \cap [\![\phi_2]\!]^a_\mu\sigma \\
[\![EX\phi]\!]^a_\mu\sigma &= \mathit{apre}([\![\phi]\!]^a_\mu\sigma) \\
[\![AX\phi]\!]^a_\mu\sigma &= \widetilde{\mathit{apre}}([\![\phi]\!]^a_\mu\sigma) \\
[\![\mu Z.\phi]\!]^a_\mu\sigma &= \mathrm{lfp}(F_a) \\
&\quad \text{where } F_a(A') = [\![\phi]\!]^a_\mu\sigma[Z/A'] \\
[\![\nu Z.\phi]\!]^a_\mu\sigma &= \mathrm{gfp}(F_a) \\
&\quad \text{where } F_a(A') = [\![\phi]\!]^a_\mu\sigma[Z/A']
\end{aligned}
$$

Soundness of abstraction for model checking



All this is completely standard abstract interpretation. An abstract model checking procedure follows directly (with proofs by refutation).

A Constraint based abstract domain

- Transition systems in which the transitions can be represented as a finite set of *transition rules* of the form $\bar{x}_1 \xrightarrow{c(\bar{x}_1,\bar{x}_2)} \bar{x}_2$.
- Concrete state space is a (possibly infinite) set of n-tuples $C \subseteq R^n$.
- Abstract state space is a finite partition of the reachable states of C (computed automatically)
- Each region in the partition $\{d_1, \ldots, d_n\}$ is represented by a linear constraint. Constraint c_{d_i} represents d_i.

Abstract operations expressed using constraint operations

$$pre(c'(\bar{y})) = \bigvee\{\text{proj}_{\bar{x}}(c'(\bar{y}) \wedge c(\bar{x},\bar{y})) \mid \bar{x} \xrightarrow{c(\bar{x},\bar{y})} \bar{y} \in T\}$$
$$\widetilde{pre}(c'(\bar{y})) = \neg(pre(\neg c'(\bar{y})))$$
$$\text{states}(p) = p$$
$$\alpha(c) = \{d \in A \mid \text{SAT}(c_d \wedge c)\}$$
$$\gamma(V) = \bigvee\{c_d \mid d \in V\}$$

- *SAT* can be implemented by an SMT solver. We use Yices (http://yices.csl.sri.com/) interfaced to Prolog.
- proj can be implemented by a linear constraint solver. We use The Parma Polyhedra Library.
- $(\alpha \circ pre \circ \gamma)$ and $(\alpha \circ \widetilde{pre} \circ \gamma)$ can be directly implemented with no loss of precision. Some optimisations on the composed functions can be performed (see LPAR paper).

Some Experiments

System	*Property*	A	Δ	*secs.*
Water Monitor	$AF(W \geq 10)$	5	4	0.02
	$AG(0 \leq W \wedge W \leq 12)$	5	4	0.01
	$AF(AG(1 \leq W \wedge W \leq 12))$	5	4	0.02
	$AG(W = 10 \rightarrow AF(W < 10 \vee W > 10))$	10	4	0.05
	$AG(AG(AG(AG(AG(0 \leq W \wedge W \leq 12)))))$	5	4	0.02
	$EF(W = 10)$	10	4	0.01
	$EU(W < 12, AU(W < 12, W \geq 12))$	7	4	0.04
Task Sched.	$EF(K2 = 1)$	18	12	0.53
	$AG(K2 > 0 \rightarrow AF(K2 = 0))$	18	12	0.30
	$AG(K2 \leq 1)$	18	12	0.04

The story so far ...

- Direct abstraction framework, yielding over-approximation of temporal logic semantics
- Galois connections not tied to any particular kind of abstraction (e.g. partitions)
- No need for (dual) abstract transition systems
- For constraint-based domains, direct implementation using constraint solvers and satisfiability checkers.

Modalities in program analysis

Possibly vs. definitely.

Describe values that possibly arise.

Describe values that definitely arise.

The two are related: definitely(P) $\Leftrightarrow \neg$ possibly($\neg$ P).

Over- and Under-approximations

Suppose that there is a Galois connection from 2^Q to some abstract domain A, written $\langle 2^Q, \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle A, \sqsubseteq \rangle$.

Then there is a dual Galois connection $\langle 2^Q, \supseteq \rangle \underset{\tilde{\alpha}}{\overset{\tilde{\gamma}}{\leftrightarrows}} \langle A, \sqsupseteq \rangle$.

where $\tilde{\alpha} = \sim \circ \alpha \circ \neg$ and $\tilde{\gamma} = \neg \circ \gamma \circ \sim$. (This assumes that there is a complement operator $\sim$ on the abstract domain.)

These Galois connections give over- and under-approximations (w.r.t. $\subseteq$ in 2^Q) respectively.
(See P. Cousot, SARA 2000).

Application to μ-calculus abstraction

Plugging in the dual Galois connections

- $\langle 2^Q, \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle 2^A, \subseteq \rangle$ and
- $\langle 2^Q, \supseteq \rangle \underset{\tilde{\alpha}}{\overset{\tilde{\gamma}}{\leftrightarrows}} \langle 2^A, \supseteq \rangle$,

we can derive dual abstractions of the μ-calculus as shown above, yielding over- and under-approximations respectively.

Let us call these abstractions $[\![.]\!]^{poss}$ and $[\![.]\!]^{nec}$ respectively (following Godefroid et al.).

Another approach: derivation of $[\![.]\!]^{nec}$ from $[\![.]\!]^{poss}$

As noted earlier, definitely(P) $\Leftrightarrow \neg$ possibly($\neg$ P).

Thus, starting with our previous abstract semantics $[\![.]\!]^{a}$ (which we now call $[\![.]\!]^{poss}$), we could define $[\![\phi]\!]^{nec} = \neg[\![\neg\phi]\!]^{poss}$

Then, for example,

$$
\begin{aligned}
[\![EX\phi]\!]^{nec} &= \neg[\![\neg EX\phi]\!]^{poss} \\
&= \neg[\![AX\neg\phi]\!]^{poss} \\
&= \neg\widetilde{apre}[\![\neg\phi]\!]^{poss} \\
&= \neg\alpha\ \widetilde{pre}\ \gamma[\![\neg\phi]\!]^{poss} \\
&= \neg\alpha\ \widetilde{pre}\ \gamma\neg\neg[\![\neg\phi]\!]^{poss} \\
&= \neg\alpha\neg pre\neg\ \gamma\neg\neg[\![\neg\phi]\!]^{poss} \\
&= \bar{\alpha}\ pre\ \bar{\gamma}[\![\phi]\!]^{nec}
\end{aligned}
$$

From dual abstraction to dual transition systems

Suppose we have a partition-based abstraction A and dual Galois connections

- $\langle 2^Q, \subseteq \rangle \underset{\alpha}{\overset{\gamma}{\leftrightarrows}} \langle 2^A, \subseteq \rangle$ and
- $\langle 2^Q, \supseteq \rangle \underset{\tilde{\alpha}}{\overset{\tilde{\gamma}}{\leftrightarrows}} \langle 2^A, \supseteq \rangle$,

Define two transition relations on A, called T_{may} and T_{must}. These are respectively over- and under-approximations of the concrete transition relation T.

$$\begin{aligned} pre[T_{may}] &= \alpha\ pre[T]\ \gamma \\ pre[T_{must}] &= \tilde{\alpha}\ pre[T]\ \tilde{\gamma} \end{aligned}$$

Then it also follows that

$$\begin{aligned} \widetilde{pre}[T_{may}] &= \tilde{\alpha}\ \widetilde{pre}[T]\ \tilde{\gamma} \\ \widetilde{pre}[T_{must}] &= \alpha\ \widetilde{pre}[T]\ \gamma \end{aligned}$$

From dual abstraction to dual transition systems: II

Using these four equations we can rewrite our original abstract semantics in terms of abstract transition systems. Note that *both T_{may} and T_{must} are needed to compute an over-approximation*.

$$
\begin{aligned}
[\![Z]\!]^{poss}_{\mu}\sigma &= \sigma(Z) \\
[\![p]\!]^{poss}_{\mu}\sigma &= \text{astates}(p) \\
[\![\neg p]\!]^{poss}_{\mu}\sigma &= \text{astates}(\neg p) \\
[\![\phi_1 \vee \phi_2]\!]^{poss}_{\mu}\sigma &= [\![\phi_1]\!]^{poss}_{\mu}\sigma \cup [\![\phi_2]\!]^{poss}_{\mu}\sigma \\
[\![\phi_1 \wedge \phi_2]\!]^{poss}_{\mu}\sigma &= [\![\phi_1]\!]^{poss}_{\mu}\sigma \cap [\![\phi_2]\!]^{poss}_{\mu}\sigma \\
[\![EX\phi]\!]^{poss}_{\mu}\sigma &= pre[T_{may}]([\![\phi]\!]^{poss}_{\mu}\sigma) \\
[\![AX\phi]\!]^{poss}_{\mu}\sigma &= \widetilde{pre}[T_{must}]([\![\phi]\!]^{poss}_{\mu}\sigma) \\
[\![\mu Z.\phi]\!]^{poss}_{\mu}\sigma &= \text{lfp}(F_a) \\
&\quad \text{where } F_a(A') = [\![\phi]\!]^{poss}_{\mu}\sigma[Z/A'] \\
[\![\nu Z.\phi]\!]^{poss}_{\mu}\sigma &= \text{gfp}(F_a) \\
&\quad \text{where } F_a(A') = [\![\phi]\!]^{poss}_{\mu}\sigma[Z/A']
\end{aligned}
$$

Summary and wider perspectives

- Abstract transition systems, while intuitive, can introduce unnecessary complications and provide restricted forms of abstraction.
- Abstract interpretation provides a general framework in which over- and under-approximations (possible vs. necessary analyses) are dual.
- Insights from abstract model checking can throw light on classical static analyses, such as
  - type inference (calculation of the states that *cannot possibly* reach a type error), and other runtime guarantees;
  - analysis of liveness properties vs. analysis of safety properties.