

Fire-Fighting the State-Explosion Problem with Abstraction & Symmetry

Christian Kissig

`kissig@tcs.inf.tu-dresden.de`

Univ. of Techn. Dresden - Dept. of CS

[Preliminaries] Kripke Structures

A Kripke Structure over

- a set V of variables
- a domain D for the variables
- a set AP of propositions of the form $v = d$

is defined as a tuple $\langle S, R, S_0, L \rangle$ of

- a set S of states $s : V \rightarrow D$
- a binary relation $R \subseteq S \times S$
- a set $S_0 \subseteq S$ of initial states
- for each state s a labelling $L(s) \subseteq AP$

[Preliminaries] FO Representation

First Order Representation of S_0 and R

- $s_0 \in S_0 \iff \mathcal{S}_0(s_0)$

- $(s_1, s_2) \in R \iff \mathcal{R}(s_1, s_2)$

Syntactic Eye-Candy

- $s \rightarrow_R t$ for $(s, r) \in R$

[Preliminaries] (Bi-) Simulation I

Consider two Structures M and M' given by

- $M = \langle S, R, S_0, L \rangle$ over V, D , and AP
- $M' = \langle S', R', S'_0, L' \rangle$ over V, D , and AP'

then $\mathcal{S} \subseteq S \times S'$ is a simulation between M and M' if

- $L(s') = AP' \cap L(s)$ for $(s, s') \in \mathcal{S}$
- for every $(s, s') \in \mathcal{S}$ if $s \rightarrow_R t$ then there is a $t' \in S'$ such that $s' \rightarrow_{R'} t'$ and $(t, t') \in \mathcal{S}$

\mathcal{S} is called a bisimulation if \mathcal{S}^{-1} is a simulation, too.

[Preliminaries] (Bi-) Simulation II

M is said to be simulated by M' (written $M \preceq_{\mathcal{S}} M'$) if there is a simulation $\mathcal{S} \subseteq S \times S'$ with

- for each initial state $s_0 \in S_0$ there is an initial state $s'_0 \in S'_0$ such that $(s_0, s'_0) \in \mathcal{S}$

If additionally

- \mathcal{S} is a bisimulation and
- for each initial state s'_0 there is an initial state s_0 with $(s'_0, s_0) \in \mathcal{S}^{-1}$

then M' is said to bisimulate M (written $M \simeq_{\mathcal{S}} M'$).

[Preliminaries] (A)CTL*

CTL* contains

- Path Quantifiers: $\mathbf{A}\phi$ (for all paths) and $\mathbf{E}\phi$ (for some path)
- Temporal Operators : $\mathbf{X}\phi$ (next state), $\mathbf{F}\phi$ (eventually), $\mathbf{G}\phi$ (always)
- Boolean Connectives : $p \in AP$ (atomic propositions), $\neg\phi$, $\phi \wedge \psi$, $\phi \vee \psi$
- Composite Connectives : $\phi\mathbf{U}\psi$ (until), $\phi\mathbf{R}\psi$ (before)

ACTL*

- contains **only** Universal Quantifiers : $\mathbf{A}\phi$ and $\mathbf{G}\phi$
- allows only for primitive negation : $\neg p$

[Preliminaries] Properties of \preceq and \simeq

Theorem 1 *If $M \preceq M'$ then for any **ACTL*** formula ϕ we obtain $M' \models \phi \Rightarrow M \models \phi$.*

Theorem 2 *If $M \simeq M'$ then for any **CTL*** formula ψ we obtain $M' \models \psi \iff M \models \psi$*

For proofs I refer to the Lecture on “Equivalence and Preorders between Structures” given by Vu Duc Lam next week.

[Example] Structure

The Example :

- crossing of two streets
- with a traffic light and a pedestrian signal each

Constructing a Kripke Model :

- Variables : t_1, \dots, t_4 for traffic lights, p_1, \dots, p_4 for pedestrian signals
- Domain : $\{green, yellow, red\}$
- States : Valuation of Variables w.r.t. the Domain

[Example] FO Representation I

The **next-state relation** R

$\mathcal{R}(s_1, s_2)$ iff

$(s_1(t_1) = \textit{green} \Rightarrow (s_2(t_1) = \textit{yellow} \wedge s_2(p_1) = \textit{red})) \wedge$

$(s_1(t_1) = \textit{yellow} \Rightarrow (s_2(t_1) = \textit{red} \wedge s_2(p_1) = \textit{green})) \wedge$

$(s_1(t_1) = \textit{red} \Rightarrow (s_2(t_1) = \textit{green} \wedge s_2(p_1) = \textit{red})) \wedge$

\vdots

$(s_1(t_4) = \textit{green} \Rightarrow (s_2(t_4) = \textit{yellow} \wedge s_2(p_4) = \textit{red})) \wedge$

$(s_1(t_4) = \textit{yellow} \Rightarrow (s_2(t_4) = \textit{red} \wedge s_2(p_4) = \textit{green})) \wedge$

$(s_1(t_4) = \textit{red} \Rightarrow (s_2(t_4) = \textit{green} \wedge s_2(p_4) = \textit{red}))$

or

every traffic light: $\textit{green} \Rightarrow \textit{yellow} \Rightarrow \textit{red} \Rightarrow \textit{green} \Rightarrow \dots$

every pedestrian signal: $\textit{red} \Rightarrow \textit{red} \Rightarrow \textit{green} \Rightarrow \textit{red} \Rightarrow \dots$

[Example] FO Representation II

Initial States S_0

$\mathcal{S}_0(s)$ iff

$s = \{t_1, t_3 \mapsto \text{green}; p_1, p_3 \mapsto \text{red}, t_2, t_4 \mapsto \text{red}, p_2, p_4 \mapsto \text{green}\} \vee$

$s = \{t_1, t_3 \mapsto \text{red}; p_1, p_3 \mapsto \text{green}, t_2, t_4 \mapsto \text{green}, p_2, p_4 \mapsto \text{red}\}$

or

a state is initial if exactly two opposite traffic lights are green and pedestrian can cross the other street

[Abstraction] Idea

- a state is a valuation $s : V \rightarrow D$
- to reduce state space
 - either reduce V : Cone-of-Influence Reduction
 - or reduce D : Data Abstraction
- Cone-of-Influence Reduction: reduce Variables to necessary ones
- Data Abstraction: Map Values to abstract Domain (of smaller Cardinality)

[Abstraction] Cone-of-Influence

Variable Dependency :

- S_0 and R can be expressed by FO-formulae, s.t.
 $v = f(V')$
- variable dependency, i.e. $depend(v) \subseteq V'$
- $depend^*$ is transitive closure of $depend$

Cone of Influence :

- for ϕ a CTL* formula with atomic propositions over variables V_ϕ define the cone-of-influence $C(\phi)$ as

$$C(\phi) = V_\phi \cup \bigcup_{v \in V_\phi} depend^*(v)$$

[Abstraction] Reduced Structure

Let $M = \langle S, R, S_0, L \rangle$ be a Structure and the set $C = \{v_1, \dots, v_k\}$ is a Cone of Influence,

then the **Reduced Structure** is given by

- $S^C = \{s^C \mid s \in S \wedge s^C = s|_C\}$
- $s^C \rightarrow_{R^C} t^C$ if there are states $s \rightarrow_R t$ with $s^C = s|_C$ and $t^C = t|_C$
- $L(s^C) = \{(v = d) \mid \exists s \in S. (s^C = s|_C \wedge (v = d) \in L(s) \wedge v \in C)\}$
- $S_0^C = \{s_0^C \mid s_0 \in S_0 \wedge s_0^C = s_0|_C\}$

[Abstraction] Semantical Properties

Theorem 3 *Let $M = \langle S, R, S_0, L \rangle$ be a Kripke Structure and $M^C = \langle S^C, R^C, S_0^C, L^C \rangle$ its Cone-of-Influence Reduction, then for every CTL* formula ϕ over variables in C*

$$M^C \models \phi \iff M \models \phi$$

Proof: By construction of a Bisimulation \mathcal{B}

$$(s, s^C) \in \mathcal{B} \iff s^C = s|_C$$

[Example] Crossing - Col

Higher Order Description of the Crossing:

$$f(t_i) = \begin{cases} \text{green} : t_i = \text{red} \\ \text{yellow} : t_i = \text{green} \\ \text{red} : t_i = \text{yellow} \end{cases}$$

and

$$f(p_i) = \begin{cases} \text{green} : t_i = \text{yellow} \\ \text{red} : t_i = \text{green} \vee t_i = \text{red} \end{cases}$$

Variable Dependencies :

$$\text{depend} = \{t_i, p_i \mapsto \{t_i\} \mid i \leq 4\}$$

[Example] Crossing - Col

Property : $F = AG.\neg((t_1 = green) \wedge (p_1 = green))$

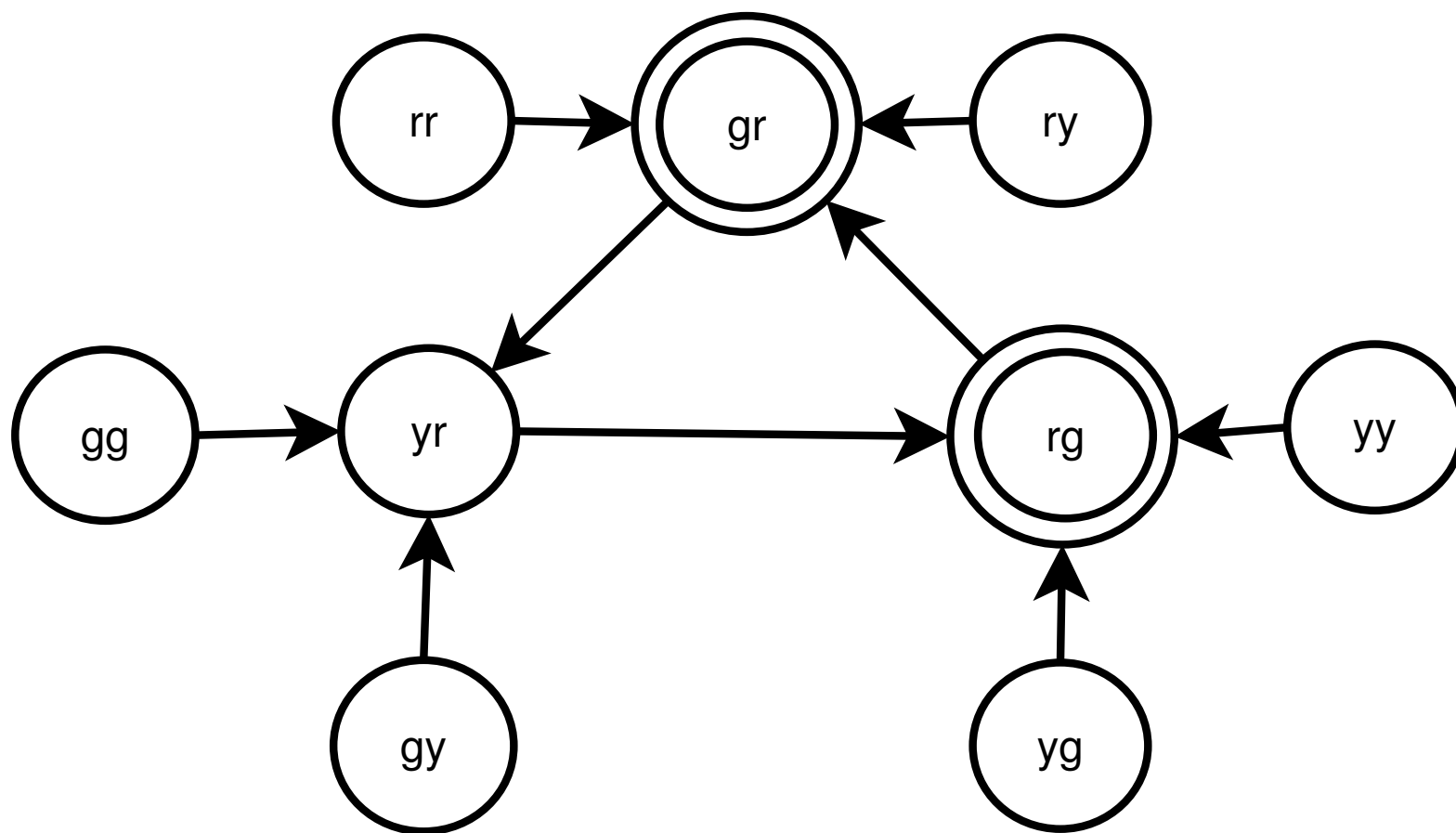
- Variables in F : $V = \{t_1, p_1\}$
- Cone of Influence : $C(V) = \{t_1, p_1\}$

Reduced State Space :

$$S = \begin{array}{ccc} (t_1 \mapsto g, p_1 \mapsto g) & (t_1 \mapsto g, p_1 \mapsto y) & (t_1 \mapsto g, p_1 \mapsto r) \\ (t_1 \mapsto y, p_1 \mapsto g) & (t_1 \mapsto y, p_1 \mapsto y) & (t_1 \mapsto y, p_1 \mapsto r) \\ (t_1 \mapsto r, p_1 \mapsto g) & (t_1 \mapsto r, p_1 \mapsto y) & (t_1 \mapsto r, p_1 \mapsto r) \end{array}$$

- Reduction from 6561 to 9 States

[Example] Crossing - CoI



[Abstraction] Data Abstraction

Underlying Idea :

- for a state $s : V \rightarrow D$ define
- a (surjective) abstraction function $h : D \rightarrow A$

Equivalence Relation \sim_h for h :

- $s \sim_h t \iff h \circ s = h \circ t$

Reduce Structure to Quotient Structure w.r.t. \sim_h

[Abstraction] Approximation

For $M = \langle S : V \Rightarrow D, R, S_0, L \rangle$ a Structure and $h : D \rightarrow A$ an abstraction function, $\hat{M} = \langle \hat{S} : V \Rightarrow A, \hat{R}, \hat{S}_0, \hat{L} \rangle$ is its **Approximation** iff

- $\exists s_0 (\hat{s}_0 = h \circ s_0 \wedge \mathcal{S}_0(s_0)) \Rightarrow \hat{s}_0 \in \hat{S}_0$
- $\exists s_1, s_2. (s_1 \rightarrow_R s_2 \wedge \hat{s}_1 = h \circ s_1 \wedge \hat{s}_2 = h \circ s_2) \Rightarrow \hat{s}_1 \rightarrow_{\hat{R}} \hat{s}_2$
- $\forall \hat{s} \in \hat{S}. \exists s \in S. \hat{L}(\hat{s}) = \{(v = a) \mid \exists (v = d) \in L(s). a = h(d)\}$

We write $M \sqsubseteq \hat{M}$.

[Abstraction] Minimal Approximation

Let $M = \langle S, R, S_0, L \rangle$ be a Structure and $M^{min} = \langle S^{min}, R^{min}, S_0^{min}, L^{min} \rangle$ an Approximation then M^{min} is the **Minimal Approximation** of M iff

- $S^{min} = \{s^{min} \mid \exists(s \in S).s^{min} = (h \circ s)\}$
- $S_0^{min} = \{s_0^{min} \in S^{min} \mid \exists(s_0 \in S_0).s_0^{min} = (h \circ s_0)\}$
- $R^{min} = \{(s^{min}, t^{min}) \mid \exists(s \in S, t \in S).(s^{min} = (h \circ s) \wedge t^{min} = (h \circ t) \wedge (s \rightarrow_R t))\}$
- $L^{min}(s^r) = \{v^{min} = a \mid \exists(s \in S)(v = d \in L(s).s^r = h \circ s \wedge a = h(a))\}$

[Abstraction] Semantical Properties

Theorem 4 *Let $M \sqsubseteq_h \hat{M}$ for an abstraction function $h : D \rightarrow A$ then for any ACTL* formula ϕ it holds that $\hat{M} \models \hat{\phi} \Rightarrow M \models \phi$.*

Proof: By Construction of a Simulation $\mathcal{H} \subseteq S \times \hat{S}$ as

$$(s, \hat{s}) \in \mathcal{H} \iff \hat{s} = h \circ s$$

[Abstraction] Lifting

Definition 5 (Lifting (Unary)) For ϕ , ϕ_1 , and ϕ_2 being first order formulae we define the following lifting

1. If P is a primitive proposition and s a state then

$$\mathcal{T}(P)(\hat{s}) = \exists s \in S. (\hat{s} = h \circ s \wedge P(s'))$$

$$\mathcal{T}(\neg P)(\hat{s}) = \neg \exists s \in S. (\hat{s} = h \circ s \wedge P(s'))$$

2. $\mathcal{T}(\phi_1 \wedge \phi_2) = \mathcal{T}(\phi_1) \wedge \mathcal{T}(\phi_2)$

3. $\mathcal{T}(\phi_1 \vee \phi_2) = \mathcal{T}(\phi_1) \vee \mathcal{T}(\phi_2)$

4. $\mathcal{T}(\forall x. \phi) = \forall \hat{x}. \mathcal{T}(\phi)$

5. $\mathcal{T}(\exists x. \phi) = \exists \hat{x}. \mathcal{T}(\phi)$

Lifting pushes existential Quantification inside.

[Abstraction] Lifting

Definition 6 (Lifting (Binary)) For ϕ , ϕ_1 , and ϕ_2 being first order formulae we define the following lifting

1. If R is a primitive relation and s_1, s_2 being states then

$$\mathcal{T}(R)(\hat{s}_1, \hat{s}_2) = \exists s_1 \in S, s_2 \in S.$$

$$(\hat{s}_1 = h \circ s_1 \wedge \hat{s}_2 = h \circ s_2 \wedge R(s_1, s_2)) \text{ and}$$

$$\mathcal{T}(\neg R)(\hat{s}_1, \hat{s}_2) = \neg \exists s_1 \in S, s_2 \in S.$$

$$(\hat{s}_1 = h \circ s_1 \wedge \hat{s}_2 = h \circ s_2 \wedge R(s_1, s_2))$$

$$2. \mathcal{T}(\phi_1 \wedge \phi_2) = \mathcal{T}(\phi_1) \wedge \mathcal{T}(\phi_2)$$

$$3. \mathcal{T}(\phi_1 \vee \phi_2) = \mathcal{T}(\phi_1) \vee \mathcal{T}(\phi_2)$$

$$4. \mathcal{T}(\forall x.\phi) = \forall \hat{x}.\mathcal{T}(\phi)$$

$$5. \mathcal{T}(\exists x.\phi) = \exists \hat{x}.\mathcal{T}(\phi)$$

[Abstraction] Lifting

Theorem 7 For $M = \langle S, R, S_0, L \rangle$ a Structure and $\hat{M} = \langle \hat{S}, \hat{R}, \hat{S}_0, \hat{L} \rangle$ its Approximation it holds that

$$\begin{aligned} \forall \hat{s} \in \hat{S}. (\exists s \in S. \hat{s} = h \circ s \wedge \mathcal{S}_0(s)) &\Rightarrow \mathcal{T}(\mathcal{S}_0)(\hat{s}) \text{ and} \\ \forall \hat{s}_1 \in \hat{S}, \hat{s}_2 \in \hat{S}. (\exists s_1 \in S, s_2 \in S. \hat{s}_1 = h \circ s_1 \wedge \hat{s}_2 = h \circ s_2 \wedge \\ \mathcal{R}(s_1, s_2)) &\Rightarrow \mathcal{T}(\mathcal{R})(\hat{s}_1, \hat{s}_2) \end{aligned}$$

Proof: by structural induction over the respective formula \mathcal{S}_0 or \mathcal{R} .

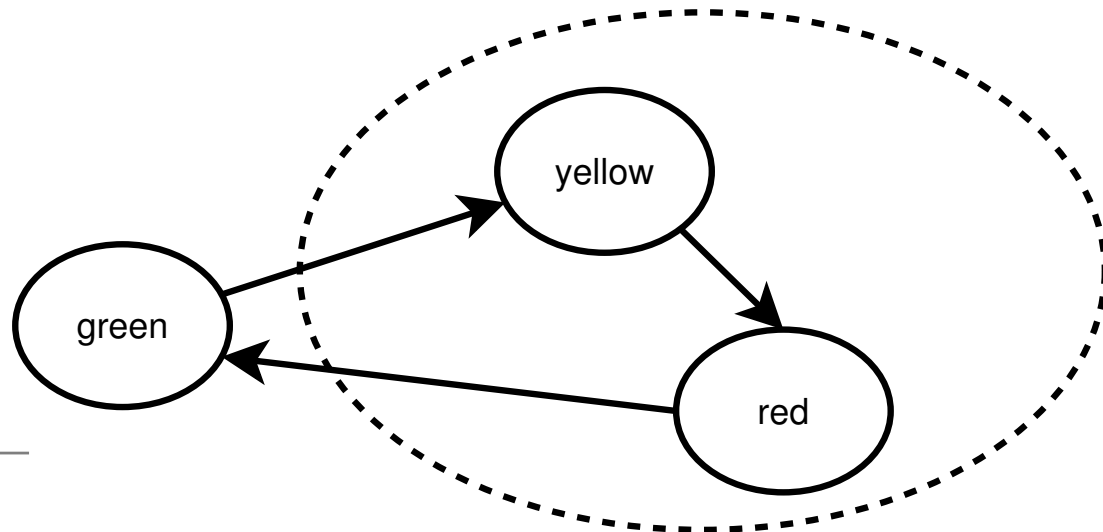
[Example] Crossing - Approx

Assume a single Traffic Light :

- $S_t = \{t \mapsto \text{green}, t \mapsto \text{yellow}, t \mapsto \text{red}\}$

- $R_t = \left\{ \begin{array}{l} (t \mapsto \text{green}) \Rightarrow (t \mapsto \text{yellow}) \\ (t \mapsto \text{yellow}) \Rightarrow (t \mapsto \text{red}) \\ (t \mapsto \text{red}) \Rightarrow (t \mapsto \text{green}) \end{array} \right\}$

- $S_{0t} = \{t \mapsto \text{green}\}$



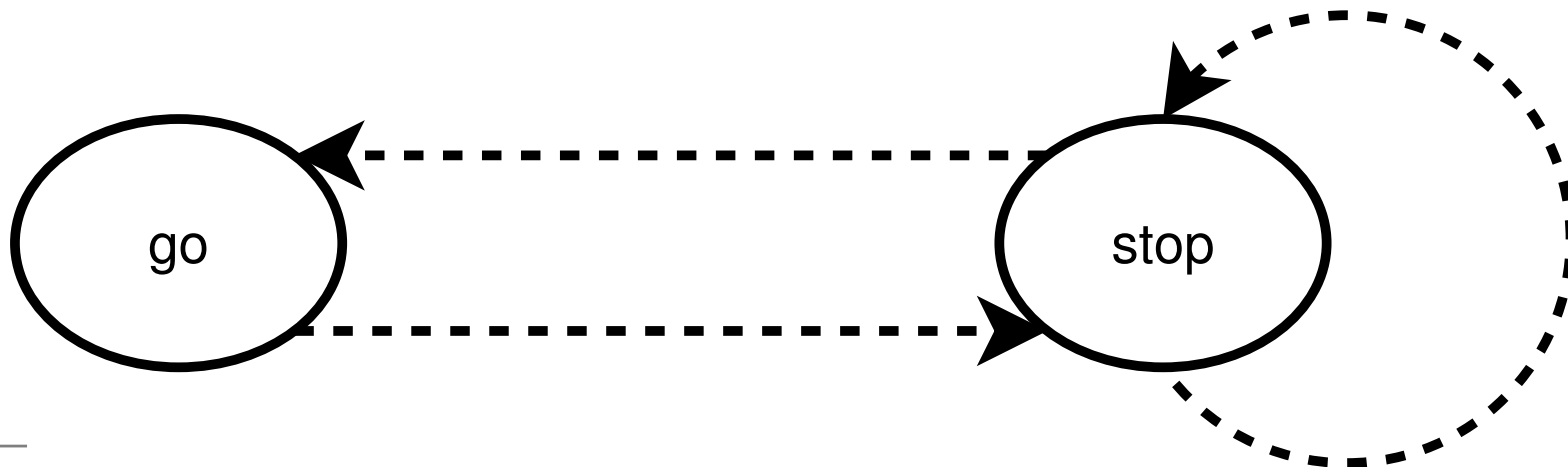
[Example] Crossing - Approx

\hat{M}_t , an Approximation for M_t :

- $S_t = \{(t \mapsto stop), (t \mapsto go)\}$

- $R_t = \left\{ \begin{array}{l} (t \mapsto stop) \Rightarrow (t \mapsto stop) \\ (t \mapsto stop) \Rightarrow (t \mapsto go) \\ (t \mapsto go) \Rightarrow (t \mapsto stop) \end{array} \right\}$

- $S_{0t} = \{t \mapsto go\}$



[Example] Crossing - Approx

FO representation for M_t :

- $\mathcal{S}_0(s) = (s(t) = \textit{green})$

$$(s_1(t) = \textit{green}) \Rightarrow (s_2(t) = \textit{yellow}) \wedge$$

- $\mathcal{R}(s_1, s_2) = (s_1(t) = \textit{yellow}) \Rightarrow (s_2(t) = \textit{red}) \wedge$

$$(s_1(t) = \textit{red}) \Rightarrow (s_2(t) = \textit{green})$$

[Abstraction] Exact Approximations

- Approximation can be computed efficiently
- Preservation of ACTL* models is rather weak

Let $M \sqsubseteq_h \hat{M}$ then \hat{M} is an **Exact Approximation** of M iff

- $\hat{s}_0 \in \hat{S}_0$ implies $\forall s \in S. (\hat{s}_0 = h \circ s \Rightarrow s \in S_0)$
- $\hat{s}_1 \rightarrow_{\hat{R}} \hat{s}_2$ implies
 $\forall s_1, s_2 \in S. (\hat{s}_1 = h \circ s_1 \wedge \hat{s}_2 = h \circ s_2 \Rightarrow s_1 \rightarrow_R s_2)$

[Preliminaries] Group Theory

A Group G is

- a set $|G|$ equipped with
- binary relation $\circ \subseteq |G| \times |G|$ (multiplication)

such that

- \circ is associative : $a \circ (b \circ c) = (a \circ b) \circ c$
- there is a neutral element e : $a \circ e = a = e \circ a$
- every a has an inverse a^{-1} : $a \circ a^{-1} = e$

[Preliminaries] Subgroups & Generators

For a Group $G = \langle |G|, \circ \rangle$ the Group $G' = \langle |G'|, \circ \rangle$ is a **Subgroup** iff

- $|G'| \subseteq |G|$ and
- G' is closed under multiplication and inverses

A set $X \subseteq |G|$ is a **Generator** of G' if G' is the smallest Subgroup of G containing X .

[Preliminaries] Permutations

- A permutation σ on a set X bijective mapping $X \rightarrow X$.
- The set of permutations σ over a set X form the group $Sym X$ defined by
 - $|Sym X|$ is the set of permutations over X
 - \circ is functional composition of permutations

[Preliminaries] Automorphisms

- Let $M = \langle S, R, S_0, L \rangle$ be a Structure
- and $\sigma : S \rightarrow S$ a Permutation on M

then σ is an Automorphism iff

- $\forall s_1, s_2 \in S. (s_1 \rightarrow_R s_2) \Rightarrow (\sigma(s_1) \rightarrow_R \sigma(s_2))$

[Preliminaries] Invariance Permutations

- Let $M = \langle S, R, S_0, L \rangle$ be a Structure
- and $\sigma : S \rightarrow S$ an **Automorphism** on M

then σ is an Invariance Permutation iff

- $\forall s \in S. L(s) = L(\sigma(s))$

(hence the name)

The set of all Invariance Permutations forms the Invariance Group, a Subgroup of the Automorphism Group.

[Symmetry] Quotient Models

For G an Invariance Group over S

- $\theta(s) = \{s' \in S \mid \exists \sigma \in G. s' = \sigma s\}$ is the **Orbit** of s

For $M = \langle S, R, S_0, L \rangle$ a Kripke Structure

$M^G = \langle S^G, R^G, S_0^G, L^G \rangle$ is a **Quotient Model** if

- $S^G = \{\theta(s) \mid s \in S\}$, i.e. set of orbits
- $R^G = \{(\theta(s_1), \theta(s_2)) \mid (s_1, s_2) \in R\}$
- $S_0^G = \{\theta(s_0) \mid s_0 \in S_0\}$
- $L^G(\theta(s)) = L(rep(\theta(s)))$

$rep : 2^S \rightarrow S$ is a choice function

[Symmetry] Semantical Properties

Theorem 8 *Let M be a Structure and M^G the corresponding Quotient Model for Invariance Group G then for any CTL* formula ϕ it holds $M^G \models \phi \iff M \models \phi$.*

Proof: by Construction of a Bisimulation \mathcal{B}

$$\mathcal{B}(s, s^G) \iff s^G = h \circ s$$

[Symmetry] The Orbit Problem

For a Structure $M = \langle S, R, S_0, L \rangle$ decide whether two states are in the same Orbit.

- A permutation on S can be directly proved to be an Invariance Permutation by applying the definition.
- The Invariance Group is given by a set of generating Invariance Permutations $\sigma_1, \dots, \sigma_n$.
- The Orbit Relation Θ is then given as the LFP of the following equation

$$Y(x, y) = (x = y \vee \exists z. (Y(x, z) \wedge \bigvee_i y = \sigma_i(z)))$$

s_1 and s_2 are in the same Orbit if $\Theta(s_1, s_2)$.

[Example] Crossing - Symmetry

For Crossing we pick out the following Permutation :

$\sigma(s_1) = s_2 \wedge \sigma(s_2) = s_1$ with

$s_1(t_1) = s_1(t_3) = s_2(t_2) = s_2(t_4) \wedge$

$s_2(t_1) = s_2(t_3) = s_1(t_2) = s_1(t_4) \wedge$

$s_1(p_1) = s_1(p_3) = s_2(p_2) = s_2(p_4) \wedge$

$s_2(p_1) = s_2(p_3) = s_1(p_2) = s_1(p_4) \wedge$

$\sigma(s) = s$ for all other states s .

● *Sym S* does only contain σ and *id*.

[Symmetry] Complexity Results

Theorem 9 *The Orbit Problem is in NP.*

Proof: by reduction to the Graph Isomorphism Problem.

- describe Graphs by their adjacency matrix
- turn matrix into vector by $x_{n(i-1)+j} = a_{ij}$
- each such vector is a state in a S
- two Graphs are isomorphic iff there vectors are in the same orbit for permutation relating vectors for graphs with swapped rows and columns

Conclusion

- Abstraction and Symmetry are powerful techniques
- but require heuristics, like h and $Sym S$

Thank you very much for your attention!