Equivalence Checking in Coalgebraic Expression Languages

Master Thesis Presentation

Moritz Blöcher

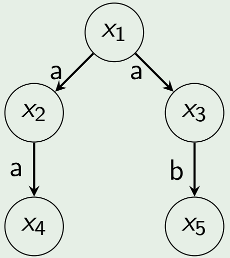
Friedrich-Alexander-University Erlangen Nürnberg

July 22, 2025

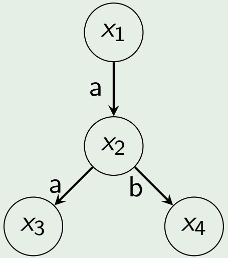
Outline

Bisimulation

Example (Graph1)



Example (Graph2)



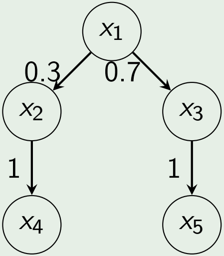
Definition (Bisimulation)

Let (S,Λ, $\rightarrow$) be a transition system then the Relation R$\subseteq S \times S$ is a Bisimulation iff forall states $(p, q) \in R$ holds:

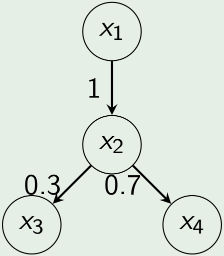
- if $p \rightarrow p'$,then there exists $q \rightarrow q'$ such that (p',q')$\in$ R
- if $q \rightarrow q'$,then there exists $p \rightarrow p'$ such that (p',q')$\in$ R

Markov Chain

Example (Graph1)

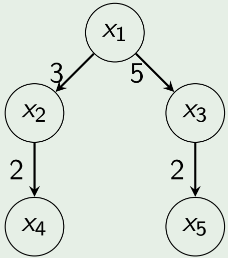


Example (Graph2)

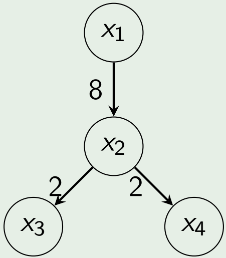


Multiraph

Example (Graph1)



Example (Graph2)



Behavioral Equivalence

Definition (behavioral equivalence)

Let (C, γ) and (D, σ) be T-coalgebras. Two states $d \in D$ and $c \in C$ are behavioral equivalent if there exists (E, ϵ) T-coalgebra and a pair of morphisms $f : (C, \gamma) \rightarrow (E, \epsilon)$ and $g : (D, \sigma) \rightarrow (E, \epsilon)$ with $f(c) = g(d)$

Definition (Λ-Bisimulations)

Let (X, ξ) and (Z, ζ) be T-coalgebras. Relation $S \subseteq X \times Z$ is called a Λ-simulation if for all predicate liftings $\lambda \in \Lambda$ and $X_1, ..., X_n \subseteq X$, xSy implies

$$\xi(x) \in \lambda(X_1, ..., X_n) \Rightarrow \zeta(y) \in \lambda(S[X_1], ..., S[X_n])$$

Iff S and S° are Λ-simulation then S is a Λ-Bisimulation

Lax Extensions

Definition (lax extension)

A relation lifting L is a lax extension if for all relations $R, R' \subseteq X \times Y$, $S \subseteq Z \times Y$ and functions $f : X \to Y$ the following holds :

$$R' \subseteq R \Rightarrow LR' \subseteq LR$$
$$LR; LS \subseteq L(R; S)$$
$$Tf \subseteq Lf$$

A lax extension is diagonal preserving if for all sets X

$$L\Delta_x \subseteq \Delta_{TX}$$

Lax Extensions

Theorem [2]

If L is a lax extension of T that preserves diagonals then L captures behavioral equivalence

[2] Marti, J., Venema, Y.: Lax extensions of coalgebra functors and their logic. J. Comput. Syst. Sci. 81(5), 880–900 (2015),Theorem 11

Strongly Expressive

Definition (singleton-preserving, strongly-expressive)

An n-ary predicate lifting $\lambda_n \in \Lambda$ is singleton-preserving if for all $x_1, ..., x_n \in X$ it holds that

$$\forall x_i \in X \Rightarrow |\lambda(\{x_1\}, \ldots, \{x_n\})| = 1$$

The set of predicate liftings Λ is strongly-expressive if

$$\forall t \in TX \; . \; \exists \lambda_n \in \Lambda, (x_1, ..., x_n) \in X.$$

$$\{t\} = \lambda_n(\{x_1\}, ..., \{x_n\})$$

Moss-Lifting

Definition (Moss-Lifting)

The predicate liftings defined by

$$\lambda = (Q^n \Rightarrow TQ \Rightarrow QT^{op})$$
$$\lambda_X(X_1, ..., X_n) = \{t \in TX | (t, \tau_{QX}(X_1, ..., X_n)) \in L(\in_X)\}$$

are called Moss-Liftings of T

For a finitary functor T and a diagonal preserving lax extension L, a set of all Moss-liftings is strongly expressive and separating

Theorem

For a seperating set Λ of monotone predicate liftings Λ-bisimulation coincides with behavioral equivalence

Characteristic fixpoint formula

Λ is strongly expressive and contains only monotone and singleton preserving predicate liftings

formulas

$$\Phi ::= v | \nu v.\Phi | L(\Phi_1, ..., \Phi_n) \qquad (v \in V, L/n \in \Lambda)$$

Characteristic fixpoint formula

characteristic system [1]

Let (X, ξ) be a T-coalgebra, let $\lambda_i/k \in \Lambda$, i =1,...,k, and let $(A_1, ..., A_k)$ be the greatest fixpoints of the equation system:

$$X_1 = \lambda_{1,X}(X_1, ..., X_k)$$

$$\vdots$$

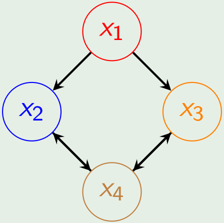
$$X_n = \lambda_{n,X}(X_1, ..., X_k)$$

Then for each i, all elements of A_i are behavioral equivalent, and for all i,j, either $A_i \cap A_j = \emptyset$ or $A_i = A_j$

[1] Ulrich Dorsch, Stefan Milius, Lutz Schröder, Thorsten Wissmann.: Predicate Liftings and Functor Presentations in Coalgebraic Expression Languages

Example

Example (Graph1)



Example

$x_1 = L(x_2, x_3)$ $x_2 = L(x_4)$

$x_3 = L(x_4)$ $x_4 = L(x_2, x_3)$

$$\nu x_1 . L(\nu x_2 . L(\nu x_4 . L(x_2, \nu x_3 . L(x_4))), \nu x_3 . L(\nu x_4 . L(\nu x_2 . L(x_4), x_3)))$$

Modal Logic

Definition (formula)

$$\phi, \psi := p \mid \bot \mid \neg\phi \mid \phi \wedge \psi \mid \heartsuit\phi$$

Definition (Semantic)

The coalgebraic Model (C, γ) containing the set of states C and map $\gamma : C \to FC$

$$c \in [\![\heartsuit\phi]\!] \Leftrightarrow \gamma(c) \in [\![\heartsuit]\!]([\![\phi]\!])$$

Relational Modal Logic

functor

$$\gamma : C \to \mathcal{P}(C)$$

modality

$$[\![\Box]\!] : Y \to \{B \in \mathcal{P}(C) | B \subseteq Y\}$$

$$[\![\Diamond]\!] : Y \to \{B \in \mathcal{P}(C) | B \cap Y \neq \emptyset\}$$

Relational Modal Logic

Moss-Lifting

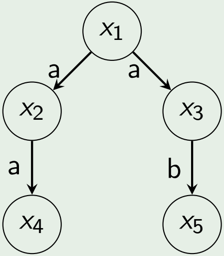
$\lambda_i(A_1, ..., A_n) = \{X \in \mathcal{P}(X) | \forall_j . \exists_{a \in A_j} a \in X \wedge \forall x \in X . \exists_j x \in A_j\}$

formula equations

$\bigwedge_{a \in A_i} \Diamond a \wedge \Box \bigvee_{a \in A_i} a$

Relational Modal Logic

Example (Graph1)



$$x_1 = \Diamond_a x_2 \wedge \Diamond_a x_3 \wedge \Box_a (x_1 \vee x_2) \wedge \Box_b \bot$$

$$x_2 = \Diamond_a x_4 \wedge \Box_a x_4 \wedge \Box_b \bot$$

$$x_3 = \Diamond_b x_5 \wedge \Box_a \bot \wedge \Box_b x_5$$

$$x_4 = \Box_a \bot \wedge \Box_b \bot$$

$$x_5 = \Box_a \bot \wedge \Box_b \bot$$

Graded Modal Logic

Definition (graded functor)

$$B(C) = \{f : C \to N \cup \infty | \text{f a function}\}$$

$$\gamma : C \to B(C)$$

modality

$$[\![\langle k \rangle]\!] : X \to \{\mu \in BC | \mu(X) > k\}$$

$$[\![[k]]\!] : X \to \{\mu \in BC | \mu(C - X) \leq k\}$$

Graded Modal Logic

Moss-Lifting

$$\lambda^{\nu}(X_1, \ldots, X_n) = \{\mu \in B_\omega(X) | \forall x \in \mathcal{P}\{X_1, ..., X_n\}. \\ \mu(x) > \nu(x) - 1; \mu(X) \leq \nu(X)\}$$

formula equation

$$\lambda^{\nu}(X_1, \ldots, X_n) \equiv (\bigwedge_{X \in \mathcal{P}\{X_1, \ldots, X_n\}} \langle (X) - 1 \rangle \bigvee_{x \in \bigcup X} x) \wedge [\mu(X)] \bot$$

Monotone Modal Logic

Definition (monotone neighbourhood functor)

$$\mathcal{N}(C) \subseteq \mathcal{P}(\mathcal{P}(C))$$
$$\mathcal{M}(C) = \{N \in \mathcal{N}(C) | N \text{ upwards closed}\}$$
$$\gamma : C \to M(C)$$

modality

$$[\![\Box]\!] : Y \to \{N \in \mathcal{M}(C) | Y \in N\}$$
$$[\![\Diamond]\!] : Y \to \{N \in \mathcal{M}(C) | \forall B \in N. B \cap Y \neq \emptyset\}$$

- No bisimulation for Neighbourhood Modal Logic
    - possible for monotone Neighbourhood

Monotone Modal Logic

Moss-lifting

$\lambda_i(A_1, ..., A_n) := \{\mathfrak{A} \in M_\omega X | \forall_i . \bigcup_j A_{i,j} \in \mathfrak{A} \text{ and } \forall B \in \mathfrak{A}.\exists_i.\forall_j.B \cap A_{i,j} = \emptyset\}$

formula equation

$\lambda_i(A_1, ..., A_n) \equiv (\bigwedge_i \Box \bigvee_j A_{i,j}) \wedge \bigwedge_\pi \Diamond \bigvee_i A_{i,\pi(I)}$
$\pi \in \{f | \forall n \in N.0 < n < |A_i| \wedge 0 < f(n) < |A_{i,n}|\}$

Behavioral Equivalence Checking

Theorem

for two states x,y the following are equivalent:

1. *x is behavioral equivalent to y*
2. *characteristic formula of x implies characteristic formula of y*
3. *y satisfies the characteristic formula of x*

Theorem [2]

every characteristic formula ϕ defines one behavioral equivalence class $[\![\phi]\!]$

Theorem[2]

Let (X,ϵ) a finite T-coalgebra and x $\in$X then there exists an expression ϕ such that x $\in [\![\phi]\!]$

[2] Ulrich Dorsch, Stefan Milius, Lutz Schröder, Thorsten Wissmann.: Predicate Liftings and Functor Presentations in Coalgebraic Expression Languages

COOL2

- Supported Modal Logics
  - Graded
  - Probabilistic
  - Coalition
  - Fixpoint variants
- Model checking
- Satisfiablity checking

Formula Size

outdegree : d , number of nodes : n , longest path : p , number of neighborhoods : nnh, members of neighborhoods : mnh

Lemma (fixpoint formula sizes)

$classic \approx (2 \cdot d)^p$
$probabilistic/graded \approx (2^n)^p$
$neighbourhood \approx ((nnh \cdot mnh) + (nnh^{mnh}))^p$

Lemma (equation system sizes)

$classic \approx n \cdot d$
$probabilistic/graded \approx n \cdot (2^n + 1)$
$neighbourhood \approx n \cdot ((nnh \cdot mnh) + (nnh^{mnh}))$

- system one formula per state
- fixpoint formula one formula per path to state
- want to use formula system where possible

Formula Equation Reasoning

Fisher-Ladner Closure

$$
\begin{aligned}
FL(p) &= \{p\} p \in Var \\
FL(\phi \wedge \psi) &= \{\phi \wedge \psi\} \cup FL(\phi) \cup FL(\psi) \\
FL(\Box\phi) &= \{\Box\phi\} \cup FL(\phi) \\
FL(\nu X.\phi) &= \{\nu X.\phi\} \cup FL(\phi[X| \to \nu X.\phi)
\end{aligned}
$$

- System only uses ν fixpoint operator (same alternating depth)
- Nesting order of fixpoint operator irrelevant for sat checking.

Formula Equation Reasoning

Definition (Parity Game moves)

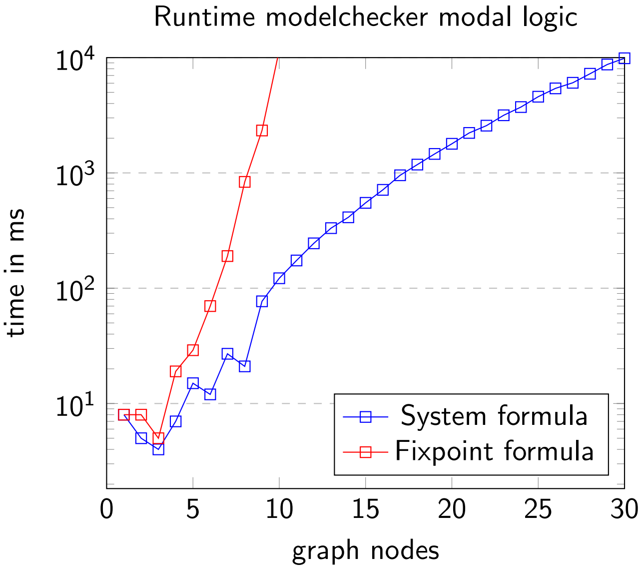
Position	Owner	Outgoing moves	Priority
$(c, \bot)$	$\exists$	$\emptyset$	0
$(c, \top)$	$\forall$	$\emptyset$	0
$(c, \phi \vee \psi)$	$\exists$	$\{(c, \phi), (c, \psi)\}$	0
$(c, \phi \wedge \psi)$	$\forall$	$\{(c, \phi), (c, \psi)\}$	0
$(c, \mu x.\phi), (c, x)$	$\exists$	$\{(c, \phi)\}$	$2 \cdot ad(x) - 1$
$(c, \nu x.\phi), (c, x)$	$\forall$	$\{(c, \phi)\}$	$2 \cdot ad(x) - 2$
$(c, \Box\phi)$	$\exists$	$\{(D, \phi) \mid \gamma(c) \in [\![\Box]\!](D)\}$	0
(D, ϕ)	$\forall$	$\{(c, \phi) \mid c \in D\}$	0

- The priority for ν is always even
- Only ν occurring in formula $\rightarrow$ Eloise wins all infinite games
- Ordering of ν nesting not relevant

Random Models

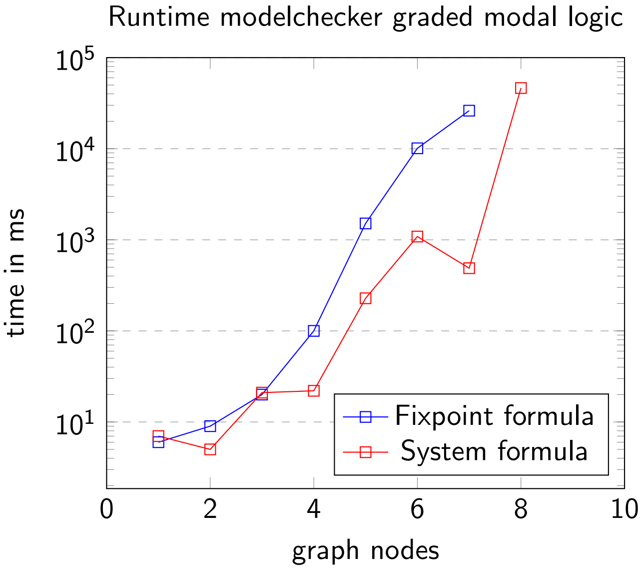
- Formula dependent on reachable nodes and degree distribution
  - No using typical random graphs e.g.
    - Erdoes Renje Models
    - Barabasi Albert Models
- Custom graph generation
  - Generate simple structure (path, tree, circle)
  - Add random edges up to defined outdegree
  - Add predicates and transition labels
  - For Multigraphs / Markov chains add weights / probabilities
- Comparisons
  - Same node
  - None bisimilar node of same graph
  - Formula of minimized graph

Runtime



Runtime modelchecker modal logic
time in ms
10^4
10^3
10^2
10^1
0
5
10
15
20
25
30
graph nodes
System formula
Fixpoint formula

Runtime



Runtime modelchecker graded modal logic
time in ms
10^5
10^4
10^3
10^2
10^1
0
2
4
6
8
10
graph nodes
Fixpoint formula
System formula

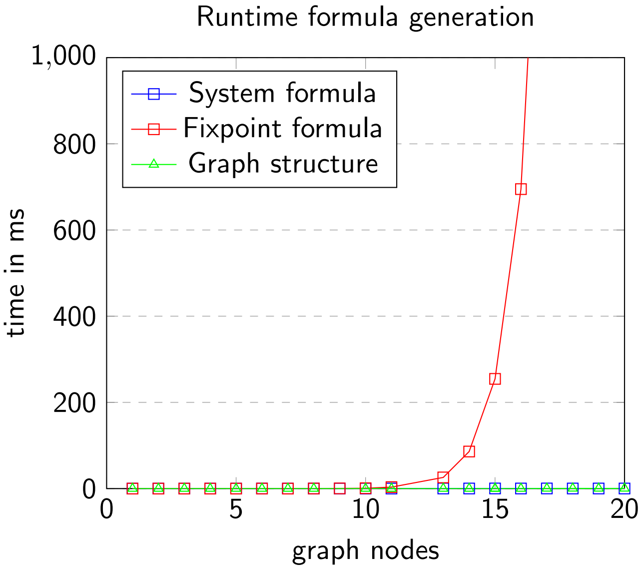
Future Work

- Optimize formula generation
- Safety games instate parity games
- Use fixpoint model checker to calculate satisfying set of states
- Scaling of small formalas on large graphs

sources

- Marti, J., Venema, Y.: Lax extensions of coalgebra functors and their logic. J. Comput. Syst. Sci. 81(5), 880–900 (2015)
- Ulrich Dorsch, Stefan Milius, Lutz Schröder, Thorsten Wissmann : Predicate Liftings and Functor Presentations in Coalgebraic Expression Languages.

Runtime


Runtime formula generation
1,000
800
600
400
200
0
time in ms
0
5
10
15
20
graph nodes
System formula
Fixpoint formula
Graph structure