Logic and Computation
in Finitely Presentable Infinite Structures

Lecture 9: Rational Structures and Modal Logic

Valentin Goranko and Sasha Rubin

ESSLLI 2006, Malaga, August 2006
Rational Transducers

Finite automata recognize relations by reading all arguments synchronously, presented by their convolution. This imposes restriction on their recognizing power.

Rational transducers (studied by Eilenberg, Elgot and Mezei, Berstel, Nivat, Johnson, Frougny and Sakarovitch, etc.) are asynchronous finite automata on pairs (tuples) of words.

Formally: \( T = \langle Q, \Sigma, q_i, F, \rho \rangle \), where:
- \( Q \) is a finite set of states,
- \( \Sigma \) is a finite alphabet,
- \( q_i \in Q \) is an initial state,
- \( F \) is a set of accepting states,
- \( \rho \subseteq Q \times (\Sigma \cup \{\varepsilon\}) \times (\Sigma \cup \{\varepsilon\}) \times Q \) is the transition relation;

Alternatively, one can take \( \rho \subseteq Q \times \Sigma^* \times \Sigma^* \times Q \).

Two views on a transducer: static, as a recognizer of a relation, and dynamic, as a transformer of words.
Rational Relations

**Rational relation**: relation, recognizable by a rational transducer.

Equivalently [Elgot and Mezei, 1965]: given an alphabet $\Sigma$, a (binary) rational relation over $\Sigma^*$ is rational iff it is a rational subset of $\Sigma^* \times \Sigma^*$, i.e., a relation generated by a rational expression from a finite subset of $\Sigma^* \times \Sigma^*$.

Example: $((0, 10)^* + (11, \varepsilon))^* \equiv (((0, 1)(\varepsilon, 0))^* + (1, \varepsilon)(1, \varepsilon))^*$.

Nivat’68: a relation $R \subseteq \Sigma^* \times \Sigma^*$ is rational iff there is a (locally) rational set $Z \subseteq \Sigma^*$ and (alphabetic) morphisms $\alpha : \Sigma^* \to X^*$ and $\beta : \Sigma^* \to Y^*$ for some alphabets $X, Y$, such that $R = \{(\alpha(w), \beta(w)) \mid w \in Z\}$.

The class of rational relations is closed under unions (but not intersections and complements), compositions, and inverses.
Rational Transition Systems

A labeled transition system \((W, \{R_a\}_{a \in A})\) is rational if:

- \(W \subseteq \Sigma^*\) is a rational language in some finite alphabet \(\Sigma\),
- Every \(R_a \subseteq \Sigma^* \times \Sigma^*\) is a rational relation in \(\Sigma\).

Some examples:

- See picture.

- **Automatic graphs**: in particular, the configuration graph of any Turing machine.

- **Counter systems**: transition systems with states labeled by vectors of integers (in e.g. binary representation), and transition relations defined on tuples of integers.

- Two important particular cases: **Presburger transition systems**, with relations definable in Presburger arithmetic, and **transition graphs of Petri nets**.
Rational graphs: some properties

Reachability in a rational graph is generally undecidable.

[Johnson, TCS’86]: testing a rational relation for any of: reflexivity, transitivity, symmetry, is not decidable (by reduction of Post Correspondence Problem).

Morvan, 2000: the query $\exists x R xx$ is undecidable, either.

Thomas, 2002: there is a rational graph with undecidable first-order theory.

Thus, the first-order theory of a rational graph is generally undecidable.

Carayol and Morvan, 2006: the first-order theory of every rational tree is decidable.

Yet, inclusion and equality of rational relations are undecidable.

So, after all, what are rational graphs good for?

For model checking of modal logic!
Basic Modal Logic

Language ML with: \( \perp, \rightarrow \), a modal operator \( \Box \), and a set of atomic propositions \( \text{AT} = \{p_0, p_1, \ldots\} \).

Formulae:

\[
\varphi = p | \perp | \varphi_1 \rightarrow \varphi_2 | \Box \varphi
\]

The other connectives are defined as usual, incl. \( \Diamond \varphi = \neg \Box \neg \varphi \).

Kripke frame: a graph (transition system) \( (W, R) \) where \( R \subseteq W^2 \).

Kripke model over a frame \( F \): a pair \( (F, V) \) where \( V : \text{AT} \rightarrow 2^W \) is a valuation. Truth of a formula \( \varphi \) at a state \( u \) of a Kripke model \( M = ((W, R), V) \):

\[
M, u \models \Box \varphi \text{ if } M, w \models \varphi \text{ for every } w \in W \text{ such that } Ruw.
\]

Resp., \( M, u \models \Diamond \varphi \) if \( M, w \models \varphi \) for some \( w \in W \) such that \( Ruw \).

The set \( \llbracket \varphi \rrbracket_M = \{u \in W \mid M, u \models \varphi\} \) is the extension of \( \varphi \) in \( M \).
Basic Modal Logic: Standard Translation

$L_0$: a FO language with $=$, a binary predicate $R$, and individual variables $\text{VAR} = \{x_0, x_1, \ldots\}$.

$L_1$: a FO language extending $L_0$ with a set of unary predicates $\{P_0, P_1, \ldots\}$, corresponding to the atomic propositions $p_0, p_1, \ldots$.

Standard translation of ML into $L_1$:

- $\text{ST}(p_i) := P_i x_0$ for every $p_i \in \text{AT}$;
- $\text{ST}(\neg \phi) := \neg \text{ST}(\phi)$.
- $\text{ST}(\phi_1 \rightarrow \phi_2) := \text{ST}(\phi_1) \rightarrow \text{ST}(\phi_2)$;
- $\text{ST}(\Box \phi) := \forall y (Rx_0 y \rightarrow \text{ST}(\phi)[x_0/y])$, where $y$ is the first variable in $\text{VAR}$ not used yet in $\text{ST}(\phi)$.

Now, for every Kripke model $M$, $w \in M$, and $\phi \in \text{ML}$:

$$M, w \models \phi \iff M, w \models_{\text{FO}} \text{ST}(\phi)(x_0 := w).$$
Modal Logic as a Fragment of Classical Logic

Satisfiability of a modal formula $\phi$ in a Kripke model $M$ is expressed in FO as:

$$M \models_{\text{FO}} \exists x \text{ST}(\phi)(x).$$

Thus, in terms of satisfiability in Kripke models, basic modal logic is a (guarded, two-variable) fragment of FO without equality.

Satisfiability of a modal formula $\phi(p_0, \ldots, p_n)$ in a Kripke frame $F$ is expressed in MSO:

$$F \models \exists P_0 \ldots \exists P_n \exists x \text{ST}(\phi)(x).$$

Thus, in terms of satisfiability in Kripke frames, modal logic is a fragment of existential MSO (EMSO).

Respectively, frame validity of a modal formula is in the Universal MSO (UMSO).

Example: Gödel-Löb formula $\Box(\Box p \rightarrow p) \rightarrow \Box p$ is valid in a frame $\langle W, R \rangle$ iff the relation $R$ is transitive and has no infinite chains.
Extensions of Basic Modal Logic

• **universal modality** $[U]$ with semantics:
  $M, u \models [U]\phi$ iff $M, w \models \phi$ for every $w \in M$, i.e. iff $M \models \phi$.

• **inverse modality** $\square^{-1}$ with semantics:
  $M, u \models \square^{-1}\phi$ iff $M, w \models \phi$ for every $w \in M$ such that $R^{-1}uw$, i.e. $Rwu$. Basic tense language.

• **transitive closure (reachability) modality** $[*]$ with semantics:
  $M, u \models [*]\phi$ iff $M, w \models \phi$ for every $w \in M$ reachable from $u$ by a finite $R$-path. Likewise, backwards reachability.

• **Until** $U$ with semantics: $M, u \models \phi U \psi$ iff $M, w \models \psi$ for some $w \in M$ reachable from $u$ by a finite $R$-path $u = u_0, \ldots, u_n$, and such that $M, u_i \models \phi$ for every $i < n$. Likewise, **Since**.

• **path quantifiers**, etc.

• **Many modal operators**, possibly indexed in some structured way, e.g. by a Kleene algebra of relations (in PDL).
Specific Important Extensions

- **LTL**: Nexttime and Until over paths in a transition system. \( F \) and \( G \) are definable.
- **CTL\(^*\)**: LTL + path quantifiers.
  - **CTL**: restricted CTL\(^*\), where all temporal operators are immediately preceded by path quantifiers.
- **modal calculus \( \mu ML \)**
Model Checking Problems in Modal Logic

On Kripke models:

- **Local model checking**: for a Kripke model $\mathcal{M}$, state $w$ in $\mathcal{M}$, and formula $\varphi$, determine whether $\mathcal{M}, w \models \varphi$.
- **Global model checking**: for a Kripke model $\mathcal{M}$ and a formula $\varphi$, determine (in some effective way) the set $\llbracket \varphi \rrbracket_{\mathcal{M}}$ of states where $\varphi$ is true.
- **Satisfiability checking**: for a Kripke model $\mathcal{M}$ and a formula $\varphi$, determine whether $\llbracket \varphi \rrbracket_{\mathcal{M}} \neq \emptyset$.

On Kripke frames:

- **Local satisfiability / validity checking**: for a Kripke frame $\mathcal{F}$, state $w$, and formula $\varphi$, determine if $\langle \mathcal{F}, V \rangle, w \models \varphi$ for some / for all valuations $V$ on $\mathcal{F}$.
- **Global satisfiability / validity checking**: for a Kripke frame $\mathcal{F}$ and a formula $\varphi$, determine the set of states $w$ in $\mathcal{F}$ such that $\langle \mathcal{F}, V \rangle, w \models \varphi$ for some / for all valuations $V$ on $\mathcal{F}$. 
Why Modal Logic on Infinite Structures?

- Modal languages are often used on infinite structures, e.g.:
  - infinite time flows.
  - transition systems of devices with **unbounded memory** (stacks, unbounded queues), or **state space** (parameterized network systems, counter systems, Petri nets, etc.)
  - Description logics on infinite domains.

- Modal languages are simple and natural frameworks to specify important properties of (infinite state) transition systems, such as **pre-** and **post-conditions**, **reachability**, **liveness**, **safety**, **fairness**, **precedence**, etc., without invoking expressive general purpose language, such as FO or MSO.

- Many natural properties of graphs can be expressed in terms of satisfiability or validity of modal formulae in relatively weak languages, e.g.: **$k$-colorability**, **existence of a kernel**, **existence of an infinite path**, **connectedness**, etc., only require the basic modal language extended with the universal modality.
Modal Logic and Symbolic Model Checking

In a TCS, 2001 paper ‘Symbolic model checking with rich assertional languages’, Kesten, Maler, Marcus, Pnueli, and Shahar formulate the following minimal requirements for an assertional language $\mathcal{L}$ to be adequate for symbolic model checking:

1. The property to be verified and the initial conditions (i.e., the set of initial states) should be expressible in $\mathcal{L}$.
2. $\mathcal{L}$ should be effectively closed under the boolean operations of negation and disjunction, and possess an algorithm for deciding equivalence of two assertions.
3. There should exist an algorithm for constructing the predicate transformer $\text{pred}$, where $\text{pred}(\phi)$ is an assertion characterizing the set of states that have a successor state satisfying $\phi$. (In modal logic terms, $\text{pred}(\phi) = \langle R \rangle \phi$.)

The basic modal logic is the minimal natural logical language satisfying these requirements, so it is good enough at least for model checking pre-conditions and post-conditions specified over regular sets of states. For more, reachability must be definable.
Model Checking Modal Logic on Finite Structures

Automata based techniques, initiated by work of Vardi and Wolper in 1980s: model checking and satisfiability checking in temporal logics of computations, such as LTL and CTL, over finite transition systems. The idea in a nutshell:

► Paths in the transition system are represented as infinite words.

► The set of all paths in a finite transition system forms an \( \omega \)-regular language, and can therefore be recognized by a Büchi automaton on infinite words.

► On the other hand, for any LTL formula a Büchi automaton can be effectively constructed, which accepts precisely those paths on which the formula is true.

► Thus, model checking and satisfiability checking can be reduced to checking inclusion and non-emptiness of \( \omega \)-regular languages.

► This method extends likewise to model checking of branching time logics by using automata on infinite trees.
Tree unfoldings of finite transition systems are simple examples of infinite state systems, where model checking can be done effectively, because every transition system is bisimilar to its unfolding, and modal logic is invariant under bisimulation.

Thus, the automata-based methods readily extend to infinite Kripke models that are effectively bisimilar to finite ones.

What to do when that is not the case?

We are looking for an effective representation of infinite Kripke models, that is robust with respect to all modal operators.
Rational Kripke models

A Kripke model \((W, \{R_a\}_{a \in A}, V)\) is rational if:

- \((W, \{R_a\}_{a \in A})\) is a rational transition system;
- \(V\) is a rational valuation, i.e. every \(V(p)\) is a rational subset of \(W\).

An example: the grid \(\mathbb{N} \times \mathbb{N}\), with atomic propositions for every row and column.
Model checking modal logic on rational models

**Proposition** If $X \subseteq \Sigma^*$ is a rational set and $R \subseteq \Sigma^* \times \Sigma^*$ is a rational relation, then the set

$$[R]X = \{w \in \Sigma^* | R(w) \subseteq X\}$$

is an effectively computable (from the automata for $X$ and $R$) rational set, too.

**Theorem**: For every rational Kripke model $(W, \{R_a\}_{a \in A}, V)$, and every modal formula $\phi$ in the modal language

$L(\{[R_a]\}_{a \in A}, \{[-R_a]\}_{a \in A}, \{[R_a^{-1}]\}_{a \in A}, \{[-R_a^{-1}]\}_{a \in A})$:

$\llbracket \phi \rrbracket_M$ is an effectively computable rational subset of $W$.

Consequently, local and global model checking in $M$ of formulae from $L_t$ are decidable.

Thus: rational Kripke models form a large and natural class of models with decidable modal logic extended with modalities over the inverse and complementary transitions.
Some questions and open problems

- What is the strongest modal language for which model checking is decidable on rational Kripke models?
- Logical characterization of rational structures, e.g. in terms of interpretations?
- Identify natural restrictions of the class of RKM where important extensions of the basic modal logic are decidable; most importantly, modal logic with reachability.
- Efficient extension of automata-based model checking techniques to rational Kripke models.
- Generalize to tree-rational Kripke models (with states encoded by finite trees, and relations recognized by rational tree transducers), and $\omega$-rational models.