Automata games for multiple-model checking

Altaf Hussain & Michael Huth
Department of Computing
Imperial College London

outline

1 Motivation:

- Modelling & analysis of multiple models

2 Models and their refinement

3 Decision problems for refinement

4 Common-refinement checks

5 Automata games

6 Related work and conclusions

Motivation: Modelling & analysis of multiple Models

Reasoning about a single model may be undesirable:

1. Federated databases: each local database interpreted as single model of data.
2. High-risk requirements of stakeholders in development project as models.
3. UML modelling: collection of message sequence charts natural subject of analysis.

Reasoning about collections of considerable practical value, e.g. in determining consistency or locating conflicts.

Contributions of paper

Given finitely many models (M_i) for $i \in I$:

1. Generalized model checking problem:

   "For a property ϕ, is there a 2-valued refinement of *all* M_i satisfying ϕ?"

   is EXPTIME-complete.

2. The decision problem:

   "Is there a 2-valued refinement of all M_i?"

   is in PTIME.

3. propose efficient way of approximating the generalized model checking problem of 1.

Basic notions

Fix unary modality $\Diamond$ and finite set AP of propositions.

- *Kripke model* K: tuple (Σ, R, L) with states Σ, transition relation $R \subseteq \Sigma \times \Sigma$, and labelling function $L : \Sigma \to \mathrm{P}(AP)$.
- *modal Kripke model* M: tuple $(\Sigma, R^a, R^c, L^a, L^c)$ where (Σ, R^a, L^a) and (Σ, R^c, L^c) are Kripke models, $R^a \subseteq R^c$, and $L^a(s) \subseteq L^c(s)$ for all $s \in \Sigma$.
- (M, s) is *pointed* modal Kripke model M with initial state s.

Intuition: R^a and R^c represent *must* and *may* transitions of the model, respectively. L^a and L^c labellings assert information that is *known* to be true, and *may* be true, repectively. The complements of R^c and L^c specify impossibilities.

2 Modal Kripke models

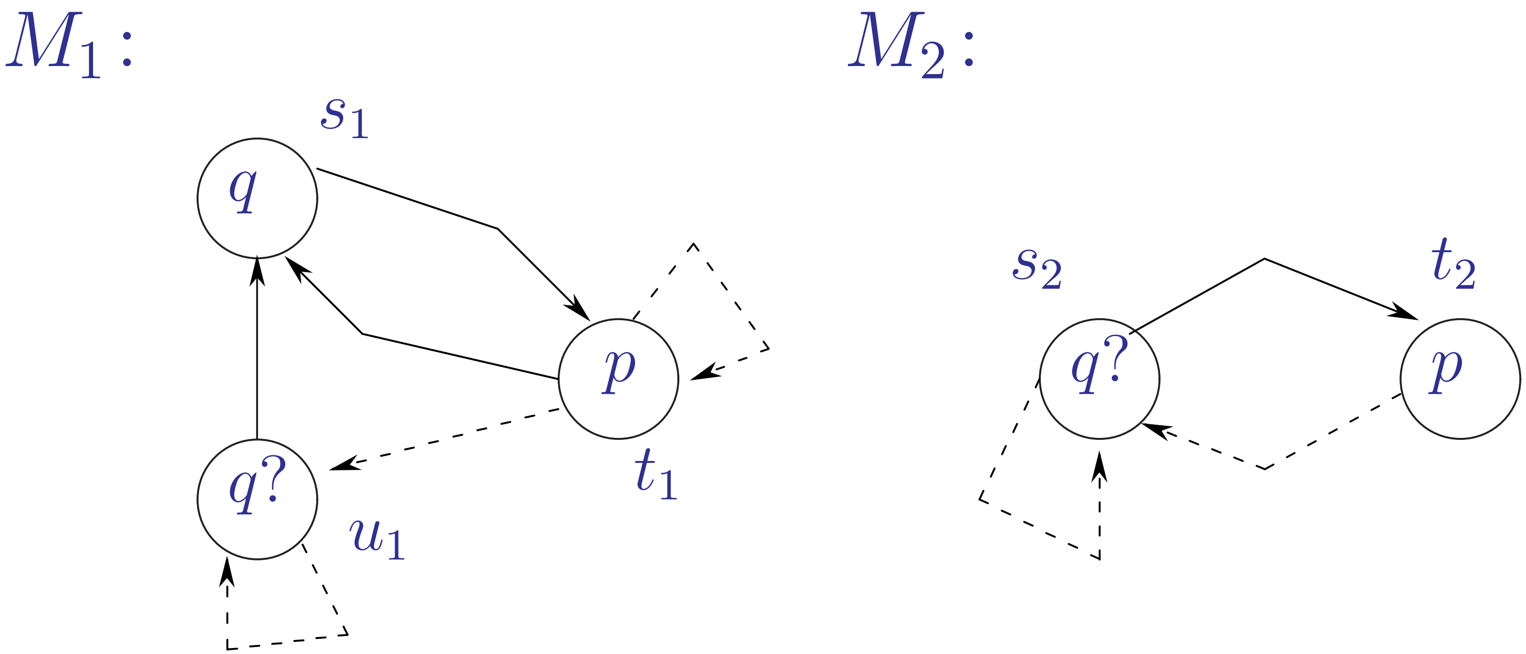


Figure 1: Two modal Kripke models. Where only $(s_1, s_2), (t_1, t_2)$ have common refinement.

Example of a Kripke model

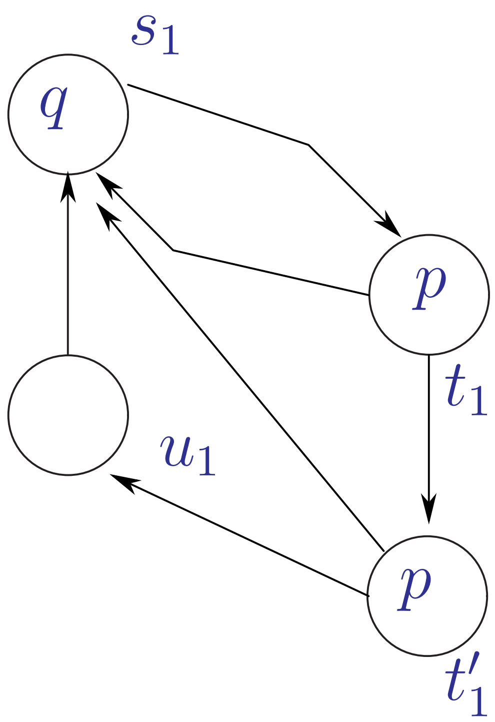


Figure 2: A Kripke model K such that (K, s_1) refines $(\mathcal{M}_1, s_1)$.

Multiple models and their decision problems

$\mathcal{C}(\mathcal{V}) = \{(N,t) \mid \forall (M,s) \in \mathcal{V} \colon (N,t) \in \mathcal{C}(M,s)\}$ = set of *common completions* of $\mathcal{V}$. For ϕ of the *modal μ-calculus*, set:

1. *Consistency:* $\mathbb{C}(\mathcal{V})$ iff all models of $\mathcal{V}$ have common completion, i.e. iff $\mathcal{C}(\mathcal{V}) \neq \{\}$.
2. *Satisfiability:* $\mathbb{S}(\mathcal{V}, \phi)$ iff there is common completion of $\mathcal{V}$ that satisfies ϕ, i.e. iff $\{(N,t) \in \mathcal{C}(\mathcal{V}) \mid t \models \phi\} \neq \{\}$.
3. *Validity:* $\mathbb{V}(\mathcal{V}, \phi)$ iff all common completions of $\mathcal{V}$ satisfy ϕ.

Complexity of the decision problems

1. Each pointed modal Kripke model (M_i, s_i) has formula $[M_i, s_i]$ of the mu-calculus such that $\mathbb{S}(\mathcal{V}, \phi)$ reduces to $SAT(\phi \wedge \bigwedge_{M_i \in \mathcal{V}} [M_i, s_i])$.

2. Decision problems $\mathbb{C}(\mathcal{V})$, $\mathbb{S}(\mathcal{V}, \phi)$, and $\mathbb{V}(\mathcal{V}, \phi)$ reducible to satisfiability checks of the modal μ-calculus over Kripke models and in EXPTIME.

Efficient consistency checking

$\mathcal{V} = \{(M_i, s_i) \mid 1 \leq i \leq k\}$, each M_i having state space Σ_i.

1. Denote product space $\prod_{i=1}^{k} \Sigma_i$ by $\Sigma_{\mathcal{V}}$, write $\mathbf{t}$ for tuples $(t_1, t_2, \ldots, t_k) \in \Sigma_{\mathcal{V}}$, and use $\mathcal{V}_{\mathbf{s}}$ to stress that s_i is the initial state in each (M_i, s_i) of $\mathcal{V}$.

2. A *common refinement witness* is a relation $W \subseteq \Sigma_{\mathcal{V}}$ such that $\mathbf{t} \in W$ implies
   (a) for all i and $q \in AP$, if $t_i \in L^a(q)$ then $t_j \in L^c(q)$ for all $j \neq i$, % "Shadow a-moves by c-moves"
   (b) for all i, if $(t_i, t_i') \in R^a$, there is $\mathbf{t}' \in W$ such that $(t_j, t_j') \in R^c$ for all $j \neq i$. % "Shadow a-moves by c-moves"

PTIME common refinement check

```
No = {};
let (bad (t, No)) = // fails clause (a) of definition:
  ((some i, j, q | t_i in L^a(q) && not t_j in L^c(q))
  ||                   // fails clause (b) of definition:
   (some (t_i,x) in R^a | all t' in S_V minus No |
      x = t'_i ==> some j | not (t_j,t_j') in R^c
  )) in
  { while (some t in S_V minus No | (bad (t, No))) {
      No = No union {t};
  }
  Yes = S_V minus No;
```

Yes returns all tuples that have common refinement

Efficient Approximations

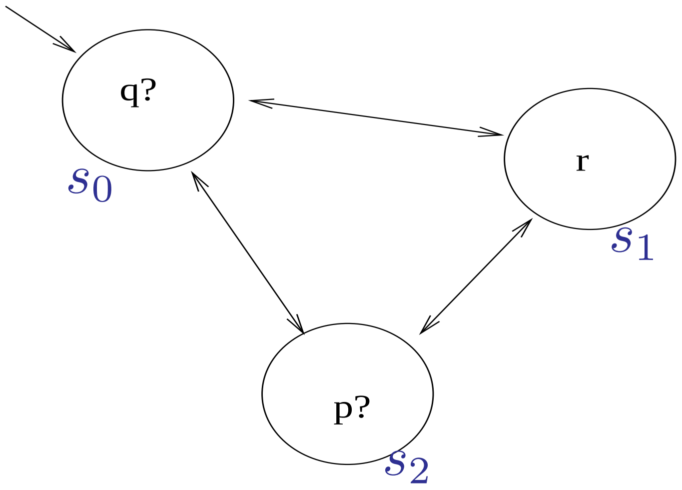
- We would like to obtain efficient approximations for $\mathbb{S}(\mathcal{V}, \phi)$ and $\mathbb{V}(\mathcal{V}, \phi)$.
- Proposal:

  1 Model merging (Larsen et al.'95, Uchitel & Chechik '04).

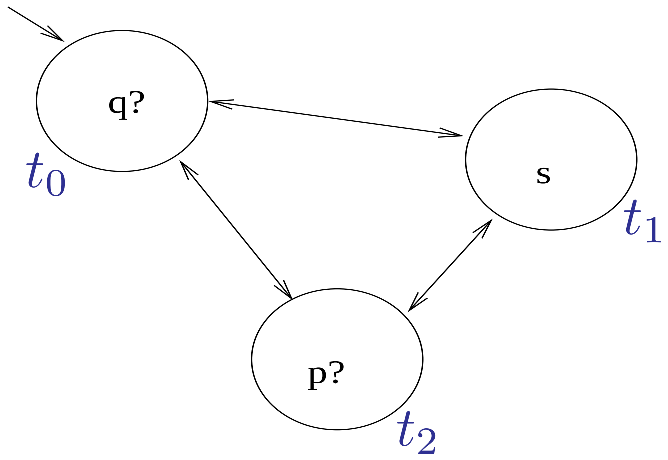
  2 Constructing "summary" models $\mathcal{V}_-$ and $\mathcal{V}_+$.

  3 Automata games.

partial Kripke models (Bruns & Godefroid '99)

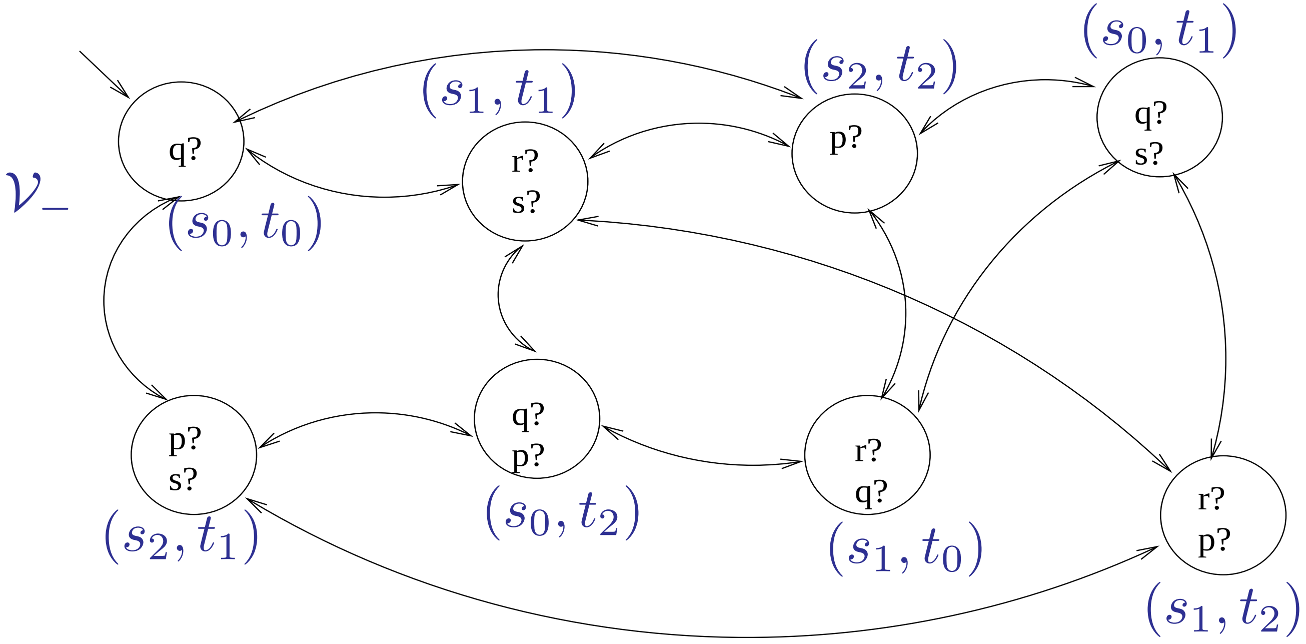


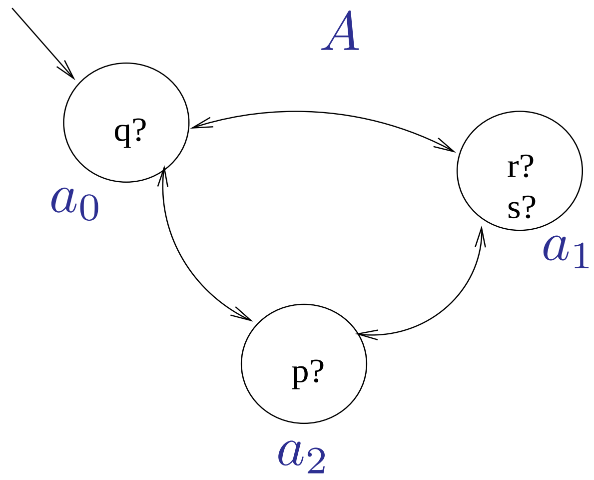
P_1



P_2

2 minimal models





Automata games

- (Dams & Namjoshi, Key Idea: 2004).

  1 Modal kripke structures and modal μ-calculus have efficient representation as *Focussed Transition Systems* (FTS).

  2 $\phi_{\mathcal{V}}$ can be expressed as such a focussed transition system.

  3 The EXPTIME-hard language inclusion $\mathcal{L}(F) \subseteq \mathcal{L}(F')$ of FTSs F and F' can be approximated in NP with a certain parity game $F \sqsubseteq F'$.

Conclusions

Studied finite sets of Modal kripke models $\mathcal{V}$ and their common refinement.

- Decision problems "Is there is common refinement of $\mathcal{V}$ satisfying ϕ?" (satisfiability) and "Does ϕ hold for all common refinements of $\mathcal{V}$?" (validity) are EXPTIME-complete for the mu-calculus.

- Gave a PTIME decision procedure for checking whether such a set $\mathcal{V}$ has common refinement (satisfiability for ϕ being $\mathtt{true}$).

- Discussed how tree automata and parity games can yield more efficient approximations for the generalized model checking problem.