Minimal Models Modulo Subset-Simulation for Modal Logics

Fabio Papacchini Renate A. Schmidt

School of Computer Science
The University of Manchester

April 12, 2013

(Minimal) Model Generation

Useful for several tasks:

- hardware and software verification
- fault analysis
- commonsense reasoning
- ...

They have been investigated for many, classical and non-classical, logics.

Minimality Criteria

Several minimality criteria has already been considered:

- domain minimality
- minimisation of a certain set of predicates
- minimal Herbrand models

Minimality Criteria

Several minimality criteria has already been considered:

- domain minimality
- minimisation of a certain set of predicates
- minimal Herbrand models

Aims

To propose a new minimality criterion for modal logics that

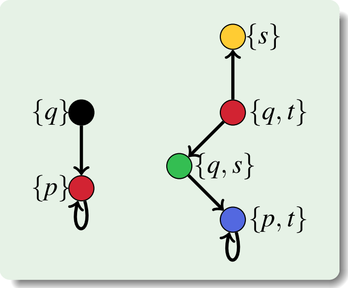
- takes in consideration the semantics of models
- is generic enough to be applied to a variety of modal logics

To propose a tableau calculus for the generation of these minimal models.

Subset-Simulation $S_{\subseteq}$

Relation between nodes of two models $M = (W, R, V)$ and $M' = (W', R, V')$ s.t.

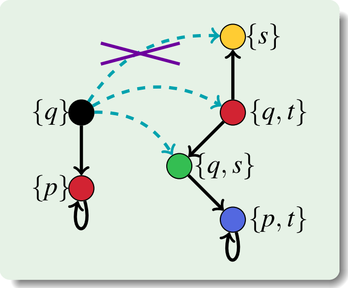
1. the subset relationship holds $(V(u) \subseteq V'(u'))$
2. successor in the first model
   $\Rightarrow$ successor in the second model
3. 1 and 2 hold for the successors of point 2



Subset-Simulation $S_\subseteq$

Relation between nodes of two models $M = (W, R, V)$ and $M' = (W', R, V')$ s.t.

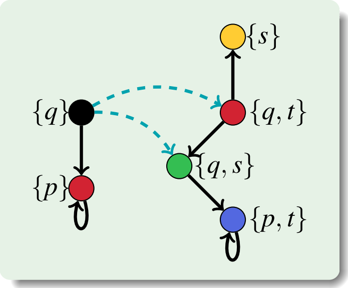
1. the subset relationship holds ($V(u) \subseteq V'(u')$)
2. successor in the first model
   $\Rightarrow$ successor in the second model
3. 1 and 2 hold for the successors of point 2



Subset-Simulation $S_\subseteq$

Relation between nodes of two models $M = (W, R, V)$ and $M' = (W', R, V')$ s.t.

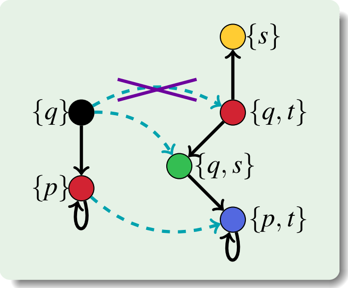
1. the subset relationship holds ($V(u) \subseteq V'(u')$)
2. successor in the first model
   $\Rightarrow$ successor in the second model
3. 1 and 2 hold for the successors of point 2



Subset-Simulation $S_{\subseteq}$

Relation between nodes of two models $M = (W, R, V)$ and $M' = (W', R, V')$ s.t.

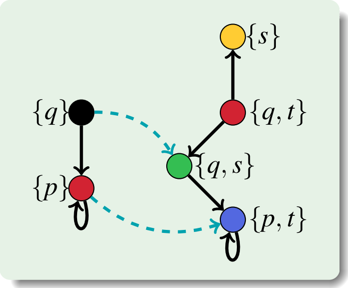
1. the subset relationship holds ($V(u) \subseteq V'(u')$)
2. successor in the first model
   $\Rightarrow$ successor in the second model
3. 1 and 2 hold for the successors of point 2



Subset-Simulation $S_\subseteq$

Relation between nodes of two models $M = (W, R, V)$ and $M' = (W', R, V')$ s.t.

1. the subset relationship holds ($V(u) \subseteq V'(u')$)
2. successor in the first model
   $\Rightarrow$ successor in the second model
3. 1 and 2 hold for the successors of point 2



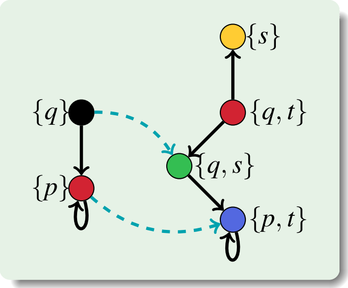
Full Subset-Simulation: for all $u \in W$ there exists some $u' \in W'$ s.t. $uS_\subseteq u'$.

Maximal Subset-Simulation: $S_\subseteq$ maximal if there is no $S'_\subseteq$ s.t. $S_\subseteq \subset S'_\subseteq$.

Subset-Simulation $S_\subseteq$

Relation between nodes of two models $M = (W, R, V)$ and $M' = (W', R, V')$ s.t.

1. the subset relationship holds ($V(u) \subseteq V'(u')$)
2. successor in the first model
   $\Rightarrow$ successor in the second model
3. 1 and 2 hold for the successors of point 2



Full Subset-Simulation: for all $u \in W$ there exists some $u' \in W'$ s.t. $uS_\subseteq u'$.

Maximal Subset-Simulation: $S_\subseteq$ maximal if there is no $S'_\subseteq$ s.t. $S_\subseteq \subset S'_\subseteq$.

We are only interested in full and maximal subset-simulations.

Minimal Models Modulo Subset-Simulation

Subset-simulation is

- reflexive
- transitive

Minimal Models Modulo Subset-Simulation

Subset-simulation is

- reflexive
- transitive

$\Longrightarrow$ a preorder

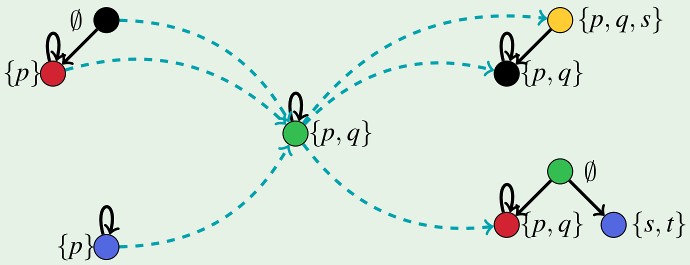
Minimal Models Modulo Subset-Simulation

Subset-simulation is

- reflexive
- transitive

$\Longrightarrow$ a preorder

Minimal models are the minimal elements of the preorder.



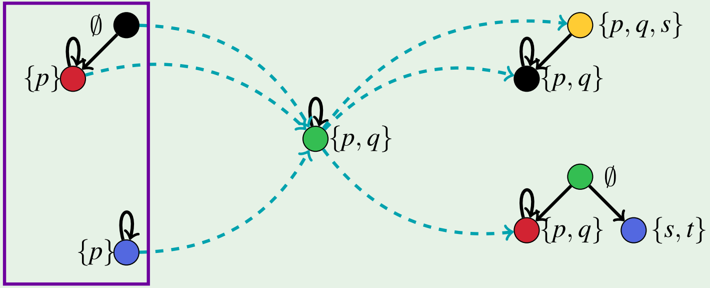
Minimal Models Modulo Subset-Simulation

Subset-simulation is

- reflexive
- transitive

$\Longrightarrow$ a preorder

Minimal models are the minimal elements of the preorder.



Minimal models

Refining Symmetric Models – Simulation

Use of simulation among symmetric minimal models allows to

- reduce the number of minimal models
- recognise bisimilar models

Symmetric w.r.t. subset-simulation:

The right model simulates the left model, but not the other way around:

Properties of the Minimality Criterion

- applied to the graph representation of models
- finite unravelled models are preferred over infinite unravelled models
- minimisation of the content of worlds
- suitable for many modal logics

Tableau Calculus

Input: a modal formula in negation normal form.

Tableau Calculus

Input: a modal formula in negation normal form.

Selection-based resolution:

- closure rule
- removes negative information from disjunctions

$$(SBR)\ \frac{u : p_1, \ldots, u : p_n \qquad u : \neg p_1 \vee \ldots \vee \neg p_n \vee \Phi_\alpha^+}{u : \Phi_\alpha^+}$$

Φ_α^+: a disjunction where no disjunct is of the form $\neg p_i$.

Tableau Calculus

Input: a modal formula in negation normal form.

Selection-based resolution:

- closure rule
- removes negative information from disjunctions

$$(SBR)\ \frac{u : p_1, \ldots, u : p_n \qquad u : \neg p_1 \vee \ldots \vee \neg p_n \vee \Phi_\alpha^+}{u : \Phi_\alpha^+}$$

Lazy clausification:

- avoids preprocessing steps
- can result in less inferences

$$(\alpha)\ \frac{u : (\phi_1 \wedge \ldots \wedge \phi_n) \vee \Phi_\alpha^+}{\begin{array}{c} u : \phi_1 \vee \Phi_\alpha^+ \\ \vdots \\ u : \phi_n \vee \Phi_\alpha^+ \end{array}}$$

Φ_α^+: a disjunction where no disjunct is of the form $\neg p_i$.

Tableau Calculus (cont'd)

Complement splitting:

- variation of the standard β rule
- detects trivially non-minimal models

$$(\beta)\ \frac{u : \mathcal{A} \vee \Phi^+}{\begin{array}{l|l} u : \mathcal{A} & u : \Phi^+ \\ u : neg(\Phi^+) & \end{array}}$$

$$\mathcal{A} ::= p \mid \Diamond\phi \mid \Box\phi$$
$$neg(\Phi^+) = \neg p_1 \wedge \ldots \wedge \neg p_n$$

Φ^+: a disjunction where no disjunct is of the form $\neg p_i$ or is a conjunction.

Tableau Calculus (cont'd)

Complement splitting:

- variation of the standard β rule
- detects trivially non-minimal models

$$(\beta)\ \frac{u : \mathcal{A} \vee \Phi^+}{\begin{array}{l|l} u : \mathcal{A} & u : \Phi^+ \\ u : neg(\Phi^+) & \end{array}}$$

$$\mathcal{A} ::= p \mid \Diamond\phi \mid \Box\phi$$
$$neg(\Phi^+) = \neg p_1 \wedge \ldots \wedge \neg p_n$$

Expansion of diamond formulae:

$$(\Diamond)\ \frac{u : \Diamond\phi}{\begin{array}{l|l|l|l} (u, u_1) : R & \ldots & (u, u_n) : R & (u, v) : R \\ u_1 : \phi & & u_n : \phi & v : \phi \end{array}}$$

v is a fresh new world

Φ^+: a disjunction where no disjunct is of the form $\neg p_i$ or is a conjunction.

Tableau Calculus (cont'd)

Complement splitting:

- variation of the standard β rule
- detects trivially non-minimal models

$$(\beta)\ \frac{u : \mathcal{A} \vee \Phi^+}{\begin{array}{l|l} u : \mathcal{A} & u : \Phi^+ \\ u : neg(\Phi^+) & \end{array}}$$

$$\mathcal{A} ::= p \mid \Diamond\phi \mid \Box\phi$$
$$neg(\Phi^+) = \neg p_1 \wedge \ldots \wedge \neg p_n$$

Expansion of diamond formulae:

$$(\Diamond)\ \frac{u : \Diamond\phi}{\begin{array}{l|l|l|l} (u, u_1) : R & \ldots & (u, u_n) : R & (u, v) : R \\ u_1 : \phi & & u_n : \phi & v : \phi \end{array}}$$

v is a fresh new world

Expansion of box formulae: the standard $\Box$ rule

Φ^+: a disjunction where no disjunct is of the form $\neg p_i$ or is a conjunction.

Properties of the Tableau Calculus

The calculus is

- refutationally sound and complete
- minimal model complete (generates all minimal models)

Properties of the Tableau Calculus

The calculus is

- refutationally sound and complete
- minimal model complete (generates all minimal models)

But it is not minimal model sound (generates also non-minimal models)!

Minimal Model Soundness – Subset-Simulation Test

Idea: incremental generation of models while closing "non-minimal" branches.

Expansion strategy: the left most branch with the least number of worlds.

Minimal Model Soundness – Subset-Simulation Test

Idea: incremental generation of models while closing "non-minimal" branches.

Expansion strategy: the left most branch with the least number of worlds.

Closure of "non-minimal" branches – Subset-Simulation Test

- Early closure of a branch: a partial model M is subset-simulated by an extracted model M', but M does not subset-simulates M'

$\Rightarrow$ close the branch from which M is extracted.

- Backward closure of branches: newly extracted model M. Compare M with the current set of minimal models and close branches accordingly.

Conclusion and Further Work

- the presented minimality criterion is semantic and suitable for many modal logics
- the calculus can be easily generalised to cover more expressive logics
- termination depends on the logic under consideration

Conclusion and Further Work

- the presented minimality criterion is semantic and suitable for many modal logics
- the calculus can be easily generalised to cover more expressive logics
- termination depends on the logic under consideration

- efficient implementation of the calculus
- study of reasonable restrictions for reducing the search space
- generalise the minimality criterion to fragments of first-order logic

Thank You!