

# A Tableau Calculus for Minimal Modal Model Generation

Fabio Papacchini<sup>1</sup> and Renate A. Schmidt<sup>2</sup>

*School of Computer Science, University of Manchester*

---

## Abstract

Model generation and minimal model generation is useful for fault analysis, verification of systems and validation of data models. Whereas for classical propositional and first-order logic several model minimization approaches have been developed and studied, for non-classical logic the topic has been much less studied. In this paper we introduce a minimal model generation calculus for multi-modal logic  $\mathbf{K}_{(m)}$  and extensions of  $\mathbf{K}_{(m)}$  with the axioms  $\mathbf{T}$  and  $\mathbf{B}$ . The calculus provides a method to generate all and only minimal modal Herbrand models, and each model is generated exactly once. A novelty of the calculus is a non-standard complement splitting rule designed for minimal model generation. Experiments show the rule has the added benefit of reducing the search space.

*Keywords:* tableaux calculus, modal logic, minimal model generation, model generation

---

## 1 Introduction

Model generation and minimal model generation is useful for fault analysis, verification of systems and validation of data models ([17,1]). For classical propositional and first-order logic several approaches have been developed for model minimization. These existing approaches can be classified as belonging to three different categories: those aiming to minimize the domain of interpretation (for example [8,10]), those aiming to minimize the interpretation of certain predicates (for example [11,12]), and those aiming to minimize the interpretation of all predicates (for example [3,13,5]).

For modal logics and related description logics minimal model generation has not been studied much. Minimal model generation has received most

---

<sup>1</sup> Email: [papacchf@cs.man.ac.uk](mailto:papacchf@cs.man.ac.uk)

<sup>2</sup> Email: [schmidt@cs.man.ac.uk](mailto:schmidt@cs.man.ac.uk)

attention for modal logics with non-monotonic operators and non-monotonic semantics, where the aim is the minimization of certain predicates (for example [6,7]). As the common modal logics can be translated into first-order logic [14], classical approaches for minimal model generation can be used to generate minimal models for modal logic formulae by using a translation approach. This approach is taken in [5], which is based on earlier work for using hyperresolution to generate Herbrand models for modal problems [9,4] and [3,13].

In this paper we focus on *the generation of minimal Herbrand models*. Though minimal Herbrand models are not domain minimal, in certain applications they tend to be more natural than domain minimal models. For example, a domain minimal model of the labelled modal formula  $Bob : \langle has\_father \rangle doctor$  is  $\{Bob : doctor, (Bob, Bob) : has\_father\}$ . This says that *Bob* is his own father. In contrast, the minimal Herbrand model is  $\{f_{\langle has\_father \rangle doctor}(Bob) : doctor, (Bob, f_{\langle has\_father \rangle doctor}(Bob)) : has\_father\}$ , where *Bob's* father is represented by the Skolem term  $f_{\langle has\_father \rangle doctor}(Bob)$  respecting the more natural meaning of the *has\_father* relation.

We introduce a *modal* approach to minimal Herbrand model generation for the multi-modal logic  $\mathbf{K}_{(m)}$  and its extensions with axioms  $\mathbf{T}$  and  $\mathbf{B}$ , that are represented by reflexive and symmetric accessibility relations. While being inspired by the PUHR approach [3] for first-order logic, our approach is based on a standard semantic labelled tableau calculus that has been adapted for generating minimal models. The calculus is designed so that the models induced by any fully expanded, open tableau branch are minimal.

Our calculus is called the *3MG calculus*, where 3MG is short for ‘minimal modal model generation’. Rather than using an explicit analytic cut rule and testing minimality by a second application of (a variation of) the calculus, as is done for example in [13,12,5,6,7], the idea of the 3MG calculus is to use complement splitting and model constraint propagation during backtracking to generate minimal models. The calculus takes as input a set of tableau clauses and returns in one run all minimal modal Herbrand models. Tableau clauses are disjunctions of labelled modal formulae and labelled relations. All models that are generated are minimal and no model is generated more than once.

The paper is structured as follows. We define the syntax and semantics and all important notions of our tableau language in Section 2, where we also give the formal definition of (minimal) modal Herbrand model. The 3MG calculus is defined in Section 3, where the generation of minimal models is illustrated with two examples. The proof of minimal model soundness and completeness is presented in Section 4. We conclude the paper with a discussion of related work and practical benefits of our non-standard complement splitting rule (Section 5), and a short summary and outlook (Section 6).

Table 1  
Modalities and their corresponding frame conditions

| $[R_i]$    | Axiom                                                                | Frame condition         |
|------------|----------------------------------------------------------------------|-------------------------|
| <b>K</b>   |                                                                      |                         |
| <b>KT</b>  | $[R_i]p \rightarrow p$                                               | reflexivity             |
| <b>KB</b>  | $p \rightarrow [R_i]\langle R_i \rangle p$                           | symmetry                |
| <b>KTB</b> | $[R_i]p \rightarrow p$<br>$p \rightarrow [R_i]\langle R_i \rangle p$ | reflexivity<br>symmetry |

## 2 Tableau Language

Our tableau calculus is designed for sets of modal formulae of propositional multi-modal logic in which the modal operators are **K**-modalities, **T**-modalities, **B**-modalities and **TB**-modalities. Semantically these modalities are characterized by no frame condition, reflexivity, symmetry and both reflexivity and symmetry, as indicated in Table 1.

A *modal formula* is a formula of the form  $\top$ ,  $\perp$ ,  $p_i$ ,  $\neg\phi$ ,  $\phi_1 \wedge \phi_2$ ,  $\phi_1 \vee \phi_2$ ,  $\langle R_i \rangle \phi$ ,  $[R_i]\phi$ , where  $\top$  and  $\perp$  are two nullary logical operators for, respectively, true and false;  $p_i$  is a propositional symbol;  $R_i$  denotes an accessibility relation;  $\neg$ ,  $\wedge$ ,  $\vee$ ,  $\langle R_i \rangle$ ,  $[R_i]$  are, respectively, the logical operators negation, conjunction, disjunction, diamond and box; and  $\phi_i$  is a modal formula.

A subformula  $\phi'$  of a modal formula  $\phi$  has *positive polarity* if  $\phi'$  is (implicitly or explicitly) in the scope of an even number of negations. A subformula  $\phi'$  of a modal formula  $\phi$  has *negative polarity* if  $\phi'$  is (implicitly or explicitly) in the scope of an odd number of negations. A modal formula  $\sim\phi$  is defined as  $\phi_1$  if  $\phi = \neg\phi_1$ , and  $\neg\phi_1$  otherwise.

The tableau calculus operates on *tableau clauses*, which are disjunctions of labelled modal formulae and labelled relations. A *labelled (modal) formula* is a pair  $u : \phi$  where  $u$  is a label,  $\phi$  is a multi-modal formula and the components of the pair are divided by the operator  $:$ . The operator  $:$  is assumed to have priority over all other operators. The labels are *terms* built from a supply of constants and unary function symbols. Intuitively,  $u : \phi$  means that  $\phi$  is true in the world represented by the term  $u$ . A *labelled relation*  $R_i$  is either of the form  $(u, v) : R_i$  or  $(u, v) : \neg R_i$ , where  $u$  and  $v$  are terms. Intuitively,  $(u, v) : R_i$  means that there is a relation  $R_i$  between  $u$  and  $v$ , while  $(u, v) : \neg R_i$  means that there is no relation  $R_i$  between  $u$  and  $v$ .

Formally, *tableau clauses* are defined by the following Backus-Naur Form production rule:

$$TC ::= \top \mid \perp \mid u : \phi \mid (u, v) : R_i \mid (u, v) : \neg R_i \mid TC \vee TC.$$

A *positive tableau literal* is a tableau clause of the form  $u : p_i$ ,  $u : \langle R_i \rangle \phi$  or a positive labelled relation  $(u, v) : R_i$ . A *negative tableau literal* is a tableau

Table 2  
Semantics of tableau formulae. Note:  $u, v \in W_{\mathcal{U}}$ ,  $\phi$  and  $\phi_i$  denote modal formulae,  $\Delta_i$  denotes tableau clauses

|                                                                                                                                                      |                                                                                     |
|------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|
| $I \models u : p_i$ iff $u : p_i \in I$                                                                                                              | $I \models (u, v) : R_i$ iff $(u, v) : R_i \in I$                                   |
| $I \not\models \perp$                                                                                                                                | $I \models \top$                                                                    |
| $I \not\models u : \perp$                                                                                                                            | $I \models u : \top$                                                                |
| $I \models u : \neg\phi$ iff $I \not\models u : \phi$                                                                                                | $I \models (u, v) : \neg R_i$ iff $I \not\models (u, v) : R_i$                      |
| $I \models u : (\phi_1 \vee \phi_2)$ iff $I \models u : \phi_1$ or $I \models u : \phi_2$                                                            | $I \models \Delta_1 \vee \Delta_2$ iff $I \models \Delta_1$ or $I \models \Delta_2$ |
| $I \models u : [R_i]\phi$ iff for every $v$ if $(u, v) : R_i \in I$ then $I \models v : \phi$                                                        |                                                                                     |
| $I \models u : \langle R_i \rangle \phi$ iff $(u, f_{\langle R_i \rangle \phi}(u)) : R \in I$ and $I \models f_{\langle R_i \rangle \phi}(u) : \phi$ |                                                                                     |

clause of the form  $u : \neg p_i$ ,  $u : [R_i]\phi$  or a negative labelled relation  $(u, v) : \neg R_i$ . A *tableau atom* is a positive tableau literal of the form  $u : p_i$  or  $(u, v) : R_i$ . We use the symbol  $\mathcal{P}$  for positive tableau literals, the symbol  $\Delta$  for tableau clauses, and  $\Delta^+$  for tableau clauses consisting only of positive tableau literals.

As our aim is to generate Herbrand models, we focus our attention on defining the notions of modal Herbrand interpretation and modal Herbrand model. It is however not difficult to extend the definition to the more general case, and showing through a specialization of the Herbrand theorem that each modal Herbrand interpretation is a standard interpretation.

Given a set of tableau clauses  $N$ , let  $W_{\mathcal{U}}$  be the set of all terms built from a supply of unary function symbols of the form  $f_{\langle R_i \rangle \phi_i}$  and  $f_{\langle R_i \rangle \sim \phi_i}$ , and the terms appearing in  $N$ . The notation indicates that  $f_{\langle R_i \rangle \phi_i}$  is uniquely associated with subformulae  $\langle R_i \rangle \phi_i$  of a labelled  $\phi$  in  $N$  with positive polarity, and  $f_{\langle R_i \rangle \sim \phi_i}$  is uniquely associated with subformulae  $[R_i] \phi_i$  of a labelled  $\phi$  in  $N$  with negative polarity. The set  $W_{\mathcal{U}}$  is the *modal Herbrand universe* for  $N$ .

The modal Herbrand semantics of tableau clauses is given by a modal Herbrand interpretation  $I$ . A *modal Herbrand interpretation*  $I$  for a tableau clause  $\Delta$  is a possibly empty set of positive tableau atoms, with all terms occurring in it belonging to  $W_{\mathcal{U}}$ . Truth in a modal Herbrand interpretation  $I$  is inductively defined in Table 2.

If a set of tableau clauses  $N$  is true in a modal Herbrand interpretation  $I$  then  $I$  is said to be a *modal Herbrand model* for  $N$ .

A property that follows directly from the definition is the following. For any interpretation  $I$ ,

$$I \models u : (\phi_1 \vee \phi_2) \text{ iff } I \models u : \phi_1 \vee u : \phi_2. \quad (1)$$

Herbrand interpretations as defined above can be conveniently ordered by the subset relation. Let  $I$  and  $I'$  be two modal Herbrand interpretations. If  $I \subseteq I'$ , then we write  $I \leq I'$ . Given a set of tableau clauses  $N$  and a modal Herbrand model  $I$  of  $N$ ,  $I$  is a *minimal modal Herbrand model* of  $N$  iff for every other modal Herbrand model  $I'$  of  $N$ , if  $I' \leq I$  then  $I = I'$ .

For example, the minimal modal Herbrand models for the tableau

clause  $w : (p_1 \wedge (\langle R_1 \rangle p_2 \vee p_3))$  in a multi-modal logic  $\mathbf{K}_{(m)}$  frame are  $I_1 = \{w : p_1, w : p_3\}$  and  $I_2 = \{w : p_1, f_{\langle R_1 \rangle p_2}(w) : p_2, (w, f_{\langle R_1 \rangle p_2}(w)) : R_1\}$ . In this case  $I_3 = \{w : p_1, w : p_3, f_{\langle R_1 \rangle p_2}(w) : p_2, (w, f_{\langle R_1 \rangle p_2}(w)) : R_1\}$  is also a modal Herbrand model of the tableau clause under consideration, but  $I_3$  is not minimal, because it is a supermodel of at least one of the other models, in fact, of both of them.

### 3 Minimal Modal Model Generation Calculus

The input of the 3MG calculus is a set of tableau clauses such that conjunction appears in a modal formula only in the scope of a diamond operator.

Given a set of labelled modal formulae, to obtain the required input we apply a clausal normal form transformation to the labelled modal formulae in the usual way, with the addition of box miniscoping. Box miniscoping is the exhaustive application of the rule  $[R_i](\phi_1 \wedge \phi_2) \Rightarrow [R_i]\phi_1 \wedge [R_i]\phi_2$ , that is, the box operator is distributed as far as possible over conjunctions. This ensures that in a modal formula a conjunction may appear only in the scope of a diamond operator, not a box operator.

For example, consider the labelled formula  $w : (p_2 \vee [R_1](p_1 \vee (p_2 \wedge \langle R_2 \rangle (p_1 \vee (p_2 \wedge p_3))))))$ . Its conjunctive normal form is (2), and the input to the calculus is the set (3).

$$w : (p_2 \vee [R_1](p_1 \vee p_2)) \wedge (p_2 \vee [R_1](p_1 \vee \langle R_2 \rangle ((p_1 \vee p_2) \wedge (p_1 \vee p_3)))) \quad (2)$$

$$\{ w : (p_2 \vee [R_1](p_1 \vee p_2)), \quad (3) \\ w : (p_2 \vee [R_1](p_1 \vee \langle R_2 \rangle ((p_1 \vee p_2) \wedge (p_1 \vee p_3)))) \}$$

The 3MG calculus consists of the six expansion rules and the model constraint propagation rule listed in Table 3.

The  $(\mathbf{T})^i$  rule accommodates the  $\mathbf{T}$  axiom in the calculus, that is, it expresses the reflexivity property for relations that are known to be reflexive. The rule is necessarily different from the rule commonly used in other tableau calculi, because terms appearing in a clause generated by the model constraint propagation rule or the negation of a diamond formula may not appear in any other tableau literals. In this case the  $(\mathbf{T})^i$  rule does not have to create any relation on them, as shown in the second example at the end of this section.

The  $(\mathbf{B})^i$  rule is the standard structural rule for accommodating the frame condition for  $\mathbf{B}$ .

The  $(\diamond)$  rule is the union of the standard  $\alpha$  rule for conjunctive formulae and the diamond rule of standard multi-modal tableaux calculi. No separate  $\alpha$  rule is needed since formulae in the input set and derived formulae are in a normal form where conjunctions can appear only immediately below a

Table 3

The rules of the 3MG calculus. Note:  $\mathcal{P}$  denotes any positive tableau literal,  $\Delta$  denotes any tableau clause,  $\Delta^+$  denotes any disjunction of positive tableau literals

| Expansion rules                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                                                                                                                                                                                                                      |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| $(\mathbf{T})^i \frac{}{(u, u) : R_i}$ <p>if <math>R_i</math> is reflexive and <math>u</math> appears in a tableau formula of the form <math>u : \phi</math>, <math>(u, v) : R_j</math> or <math>(v, u) : R_j</math> on the current branch</p>                                                                                                                                                                                                                                                      | $(\mathbf{B})^i \frac{(u, v) : R_i}{(v, u) : R_i}$ <p>if <math>R_i</math> is symmetric</p>                                                                                                                                                                                                                                                                           |
| $(\diamond) \frac{u : \langle R_i \rangle (\phi_1 \wedge \dots \wedge \phi_n)}{(u, f_{\langle R_i \rangle \phi}(u)) : R_i}$ $f_{\langle R_i \rangle \phi}(u) : \phi_1$ $\vdots$ $f_{\langle R_i \rangle \phi}(u) : \phi_n$ <p>where <math>\phi = \phi_1 \wedge \dots \wedge \phi_n</math> and <math>f_{\langle R_i \rangle \phi}</math> is the function symbol uniquely associated with <math>\langle R_i \rangle \phi</math></p>                                                                   | $(\vee)_E \frac{u : (\phi_1 \vee \dots \vee \phi_n) \vee \Delta}{u : \phi_1 \vee \dots \vee u : \phi_n \vee \Delta}$<br>$(CS) \frac{\mathcal{P}_1 \vee \dots \vee \mathcal{P}_n}{\mathcal{P}_1 \mid \mathcal{P}_2 \vee \dots \vee \mathcal{P}_n}$ <p>where <math>neg(\mathcal{P}_i)</math> stands for <math>neg(\mathcal{P}_2), \dots, neg(\mathcal{P}_n)</math></p> |
| $(\mathbf{SBR}) \frac{\begin{array}{c} u_1 : p_1 \quad \dots \quad u_n : p_n \\ (v_1, w_1) : R_{m_1} \quad \dots \quad (v_m, w_m) : R_{m_m} \\ (s_1, t_1) : R_{j_1} \quad \dots \quad (s_j, t_j) : R_{j_j} \\ u_1 : \neg p_1 \vee \dots \vee u_n : \neg p_n \vee v_1 : [R_{m_1}] \phi_1 \vee \dots \vee v_m : [R_{m_m}] \phi_m \\ \vee (s_1, t_1) : \neg R_{j_1} \vee \dots \vee (s_j, t_j) : \neg R_{j_j} \vee \Delta^+ \end{array}}{(w_1 : \phi_1) \vee \dots \vee (w_m : \phi_m) \vee \Delta^+}$ |                                                                                                                                                                                                                                                                                                                                                                      |

**Model constraint propagation rule**

If  $\mathcal{B}$  is an open and fully expanded branch in a tableau derivation generated by the 3MG calculus, and  $I = \{u_1 : p_1, \dots, u_n : p_n, (v_1, w_1) : R_1, \dots, (v_m, w_m) : R_m\}$  is the (minimal) modal Herbrand model extracted from  $\mathcal{B}$ , then the following *model constraint clause*

$$u_1 : \neg p_1 \vee \dots \vee u_n : \neg p_n \vee (v_1, w_1) : \neg R_1 \vee \dots \vee (v_m, w_m) : \neg R_m$$

is added to all the branches to the right of  $\mathcal{B}$ .

diamond operator. Another important difference to common definitions found in the literature is that the diamond rule does not create a new constant, but a new Skolem term of the form  $f_{\langle R_i \rangle \phi}(u)$ .

Since the other rules of the calculus are applicable only to disjunctions of tableau literals, the  $(\vee)_E$  rule converts disjunctions of modal formulae under a specific label to disjunctions of labelled literals. The  $(\vee)_E$  rule is the only rule that does not contribute to the generated model, since it does not add any positive or negative tableau literals to the branch. The rule is justified by the property (1) in Section 2.

The  $(CS)$  rule is the complement splitting rule. Its premise is a disjunction of positive tableau literals. An application of the  $(CS)$  rule results in the creation of two branches. One of the positive tableau literals in the premise and the negation of all the other literals are added to the left branch. The

premise, with the positive tableau literal appearing on the left branch removed, is added to the right branch. Here, the negation of literals is defined by a unary function  $neg$  as follows.

$$neg(\mathcal{P}) = \begin{cases} u : \neg p_i & \text{if } \mathcal{P} = u : p_i \\ (u, v) : \neg R_i & \text{if } \mathcal{P} = (u, v) : R_i \\ (u, f_{\langle R_i \rangle \phi}(u)) : \neg R_i & \text{if } \mathcal{P} = u : \langle R_i \rangle \phi. \end{cases}$$

That is, if a positive tableau literal has the form  $u : p_i$  or  $(u, v) : R_i$ , then its negation is simply  $u : \neg p_i$  or  $(u, v) : \neg R_i$ , respectively. The negation of positive tableau literals of the form  $u : \langle R_i \rangle \phi$  needs special handling. It is not possible to negate the diamond formula as might be expected, because this could produce non-minimal models (in Section 5 we give an example). Instead we define the negation of  $u : \langle R_i \rangle \phi$  to be  $(u, f_{\langle R_i \rangle \phi}(u)) : \neg R_i$ . The intuition is that if there is a negation of a positive tableau literal, then we want to avoid the presence and the expansion of that positive literal in this branch. Following the modal Herbrand interpretation semantics, to block a specific diamond formula we can use the relation that such a diamond formula would create if expanded. For this to work it is important that the relation is uniquely associated to that diamond formula via the terms, which is achieved in our calculus through the use of Skolem functions in the way defined.

The  $(CS)$  rule is the only branching rule of the calculus, and its aim is twofold. First, it avoids the creation of a model more than once, because each branch differs from any other branch by at least one model element (tableau atom). Second, the first model extracted from the left-most branch of the tableau is minimal. These two properties are consequences of the soundness and completeness result in Section 4.

The last expansion rule is what we refer to as the  $(SBR)$  rule. The name reflects the close relationship to selection-based resolution for first-order clause logic. The  $(SBR)$  rule is the most complex rule and is the only rule that can close a branch. It may be thought of as the simultaneous application of closure rules (for labelled formulae and labelled relations) and the box rule in standard multi-modal tableau calculi. The aim of the  $(SBR)$  rule is to expand a disjunction of tableau literals where some of the tableau literals are negative iff it is necessary. This behaviour is based on our definition of minimal modal Herbrand models, in fact, such models are composed only of specific positive tableau literals. Thus, if the expansion of a tableau clause results in a clause that contains at least one negative literal then such a clause does not contribute to the model. As the box operator hides complex modal formulae, the rule does not completely avoid the generation of clauses that may contain negative tableau literals, but it tries to avoid them as much as possible (cf. Section 5).

The last rule in the calculus is the model constraint propagation rule. It is different from the other rules in that it becomes applicable once a fully expanded, open branch has been obtained. A branch is *fully expanded* if no more rules are applicable. If the current branch  $\mathcal{B}$  is open and fully expanded, then the model constraint propagation rule extracts the Herbrand model defined by the positive tableau atoms in  $\mathcal{B}$ . This Herbrand model is used to construct the model constraint clause as described in Table 3. The modal constraint clause is added to all branches to the right of the current branch. The calculus is defined in such a way—and derivations are constructed in such a way—that any model extracted from a fully expanded, open branch is a minimal Herbrand model. The minimal model constraints added during backtracking prevent the generation of non-minimal models by immediately closing branches which begin to construct super-models. If a super-model of an already extracted model is constructed in a branch, then an application of the (*SBR*) rule with the model constraint clause as main premise closes the branch.

We assume as usual that no rule is applied more than once to the same set of premises.

The 3MG calculus is *minimal model sound and complete*, in the sense that it terminates and generates all and only minimal modal Herbrand models for a set of tableau clauses. These properties of the calculus are not only due to its rules, but also due to the search strategy used during the derivation. We assume that a depth-first left-to-right expansion strategy is used. A departure from this strategy would compromise minimal model soundness and completeness of the calculus.

The expansion rules may be applied in any order without compromising minimal model soundness and completeness. A sensible order of application is:  $(\mathbf{T})^i$ ,  $(\mathbf{B})^i$ , (*SBR*),  $(\vee)_E$ ,  $(\diamond)$ , and (*CS*), the idea being to close a branch as soon as possible to avoid useless expansion, and to delay the application of the branching rule to avoid repeated application of a rule in different branches.

For a given input set  $N$  of tableau clauses the 3MG calculus derives either a closed tableau or a fully expanded, open tableau. If a closed tableau is constructed,  $N$  is unsatisfiable. If a fully expanded, open tableau is constructed,  $N$  is satisfiable, and each open branch defines a minimal modal Herbrand model. As the 3MG calculus uses a depth-first left-to-right strategy, the process could be stopped after the first fully expanded, open branch has been constructed, if we are interested in finding only one minimal model.

We conclude this section with two examples. First, the set  $\{w : [R_1][R_1]p_1, w : (\langle R_1 \rangle \neg p_2 \vee p_1), w : (p_3 \vee \langle R_1 \rangle \neg p_2)\}$  is  $\mathbf{K}_{(m)}$ -satisfiable and has two minimal Herbrand models. Figure 1 shows how these can be derived using our tableau calculus. Each formula in the derivation is numbered, the convention being that each number represents the application of a rule. The number 0 identifies the input clauses. In this example at least one applica-

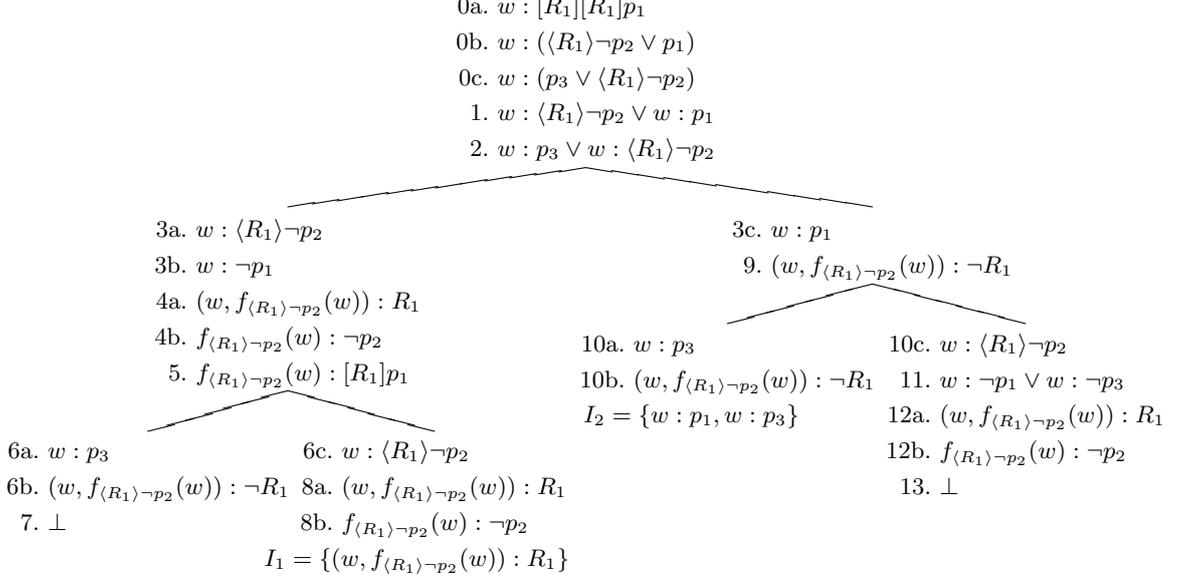


Fig. 1. Derivation for the set  $\{w : [R_1][R_1]p_1, w : (\langle R_1 \rangle \neg p_2 \vee p_1), w : (p_3 \vee \langle R_1 \rangle \neg p_2)\}$  in the multi-modal  $\mathbf{K}_{(m)}$  frame. The models returned are  $I_1 = \{(w, f_{\langle R_1 \rangle \neg p_2}(w)) : R_1\}$  and  $I_2 = \{w : p_1, w : p_3\}$

tion of each rule is shown, with the exception of the rules representing the **T** and **B** axioms. The  $(\vee)_E$  rule is applied to 0b and 0c to derive 1 and 2, to which the complement splitting rule is now applicable. The formulae numbered 3 are obtained by applying  $(CS)$  to 1. The  $(\diamond)$  rule is applied to 3a, 6c and 10c to respectively get 4, 8 and 12. As the example is simple, all  $(\diamond)$  rule applications are equivalent to applying the standard diamond rule modulo Skolem terms being introduced, which is one of the features of the calculus. The derivation shows different applications of the  $(CS)$  rule. It is possible to observe the function *neg* in operation for a diamond formula in the formulae numbered 6 and 10 that are the result of applying the  $(CS)$  rule to 2. In this case we see the *neg* function blocks the expansion of the positive tableau literal in the input. In particular, the branch finishing with 7 closes due to the contradiction between 6b and the already expanded diamond (represented by 4). The branch closes as a result of the application of the  $(SBR)$  rule to 6b and 4a. During the explanation of the  $(SBR)$  rule, we pointed out that an application of the  $(SBR)$  rule may lead to a tableau formula containing negative tableau literals. An example of this is formula 5, which is the result of applying the  $(SBR)$  rule to 0a and 4a. As the branch ending with 8 is open and fully expanded, the minimal modal Herbrand model that is given is extracted. The model constraint generated from this model is added to the only other branch as model constraint 9. In the right-most branch it is possible to note how the model constraint avoids the creation of super-models. In fact, the branch is closed by an application of the  $(SBR)$  rule to 9 and 12a.

The second example is shown in Figure 2. The input set is com-

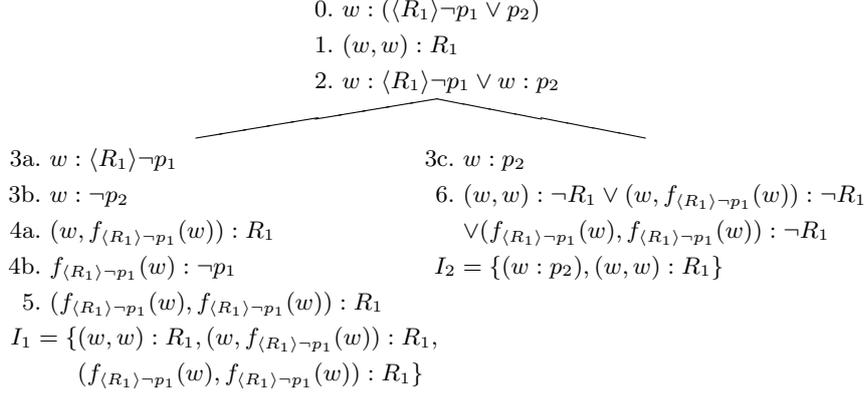


Fig. 2. Derivation for the tableau clause  $w : (\langle R_1 \rangle \neg p_1 \vee p_2)$  and  $R_1$  reflexive.

posed of the single tableau clause  $w : (\langle R_1 \rangle \neg p_1 \vee p_2)$  and  $R_1$  is assumed to be reflexive. Thanks to the side conditions of the  $(\mathbf{T})^i$  rule, the relation  $(f_{\langle R_1 \rangle \neg p_1}(w), f_{\langle R_1 \rangle \neg p_1}(w)) : R_1$  does not appear in the right branch, because the term  $f_{\langle R_1 \rangle \neg p_1}(w)$  appears only in the model constraint clause (6). A standard rule for the  $\mathbf{T}$  axiom would add the relation to the right branch, resulting in a non-minimal model.

## 4 Soundness and Completeness

Our proof of minimal model soundness and completeness is based on showing the existence of a bisimulation between the 3MG calculus and the PUHR approach ([3]). Even though there is some similarity between some of the rules, the 3MG calculus and the PUHR approach do not correspond directly to each other in the sense that a step in the modal calculus can be simulated by one or more steps of the calculus of the PUHR approach, or the other way around. We show that the two calculi are approximations of each other via a new translation of the 3MG input to first-order clauses.

This section is divided as follows: first, we recall the PUHR approach presented in [3]; second, we present a new translation from the 3MG input into first-order formulae; finally, we prove the existence of a bisimulation relationship between the 3MG calculus and a modified version of the PUHR approach on the 3MG input translated in an appropriate way.

### 4.1 The PUHR Approach

In this section we recall the definition of the PUHR approach [3], where it is called the depth-first minimal model generation procedure, used to prove minimal model soundness and completeness of the 3MG calculus in the next two sections.

In reference to the PUHR approach and first-order formulae and clauses

the notational convention is as follows. We denote first-order variables by  $x, y$ ; terms by  $t$ ; constants by  $s, u, v, w$ ; functions by  $f$ ; predicate symbols by  $P, Q, R$ ; atoms by  $A, B$ ; literals by  $L$ ; clauses by  $C, D$ ; substitutions by  $\theta, \sigma$ ; first-order formulae by  $F, E$ ; each of these could appear with subscripts or superscripts.

The notions of substitution, unifier, most general unifier, ground term, ground literal and ground clause are defined as usual. A clause is called *positive* if it contains only positive literals. By  $F[E/E']$  we denote that  $E$  is a subformula of  $F$  and that each occurrence of  $E$  is simultaneously substituted by  $E'$ .

A *Herbrand interpretation*  $H$  of a set of first-order clauses is a set of positive ground atoms. A ground atom  $A$  is true in a Herbrand interpretation  $H$  if  $A \in H$  and it is false if  $A \notin H$ . A clause  $C$  is true in a Herbrand interpretation  $H$  iff in all ground instantiations  $C\sigma$  there is at least a ground literal which is true in  $H$ . A set  $N$  of clauses is true in  $H$  iff all clauses in  $N$  are true in  $H$ . If a set  $N$  of clauses is true in an interpretation  $H$ , then such an interpretation is referred to as a *Herbrand model* of  $N$ . A Herbrand model  $H$  is said to be a *minimal Herbrand model* of a set of clauses  $N$  iff for every other Herbrand model  $H'$ , if  $H' \subseteq H$  then  $H = H'$ .

The PUHR approach is a depth-first left-to-right minimal model generation procedure. It operates on a set of range-restricted clauses having finitely many finite Herbrand models. A clause is *range-restricted* if each variable appearing in a positive literal appears also in at least one negative literal. As a consequence, if there are no negative literals in a range-restricted clause, the clause is ground. The underlying calculus of the PUHR approach is described in Table 4.1. There are two expansion rules: the positive unit hyperresolution (PUHR) rule and the complement splitting rule. In addition to these rules, the procedure involves the use of model constraint propagation during backtracking.

**Theorem 4.1** ([3]) *Let  $N$  be a finite set of range-restricted clauses. If  $N$  has finitely many finite Herbrand models, then the PUHR approach applied to  $N$  terminates, it returns all and only minimal Herbrand models of  $N$  (that is, it is complete and sound), and does not return any minimal model more than once.*

#### 4.2 A Minimal Herbrand Model Preserving Translation

Our translation from the input set of the 3MG calculus into first-order clauses is based on a variation of the standard relational translation ([14]) of modal logic to first-order logic. An important requirement for our proof is that the minimal modal Herbrand model of any input set corresponds to the minimal Herbrand model of its translation.

Table 4  
The rules of the PUHR approach

---

| <b>Expansion rules</b>                                                                                                                                                                                                                                                                                                                                                                       |                                                                                                                                                                             |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>PUHR rule:</b>                                                                                                                                                                                                                                                                                                                                                                            | $\frac{A_1, \dots, A_n \quad \neg A'_1 \vee \dots \vee \neg A'_n \vee D}{D\sigma}$                                                                                          |
| <p>where <math>A_1, \dots, A_n</math> are positive ground atoms, <math>D</math> is a positive clause and <math>\sigma</math> is such that <math>A_1 = A'_1\sigma, \dots, A_n = A'_n\sigma</math></p>                                                                                                                                                                                         |                                                                                                                                                                             |
| <b>Complement splitting:</b>                                                                                                                                                                                                                                                                                                                                                                 | $\frac{A_1 \vee \dots \vee A_n}{\begin{array}{c c c} A_1 & A_2 & A_n \\ \hline \neg A_2 & \vdots & \dots \\ \hline \vdots & \neg A_n & \\ \hline \neg A_n & & \end{array}}$ |
| <p>where <math>A_1, \dots, A_n</math> are positive ground atoms</p>                                                                                                                                                                                                                                                                                                                          |                                                                                                                                                                             |
| <b>Model constraint propagation rule</b>                                                                                                                                                                                                                                                                                                                                                     |                                                                                                                                                                             |
| <p>If <math>\mathcal{B}</math> is a finite, open and fully expanded branch of a tableau generated by the PUHR approach, and <math>H = \{A_1, \dots, A_n\}</math> is the (minimal) Herbrand model extracted from <math>\mathcal{B}</math>, then the model constraint <math>\neg A_1 \vee \dots \vee \neg A_n</math> is added to all the branches to the right of <math>\mathcal{B}</math></p> |                                                                                                                                                                             |

---

The new translation of the tableau input into a set of first-order clauses is defined using an auxiliary mapping  $St_\diamond$ .  $St_\diamond$  is defined as follows:

$$\begin{aligned}
 St_\diamond(x, \top) &= \top & St_\diamond(x, \perp) &= \perp \\
 St_\diamond(x, p_i) &= P_i(x) & St_\diamond(x, \neg\phi) &= \neg St_\diamond(x, \phi) \\
 St_\diamond(x, \phi_1 \vee \phi_2) &= St_\diamond(x, \phi_1) \vee St_\diamond(x, \phi_2) & St_\diamond(x, \langle R_i \rangle \phi) &= R_i(x, f_{\langle R_i \rangle \phi}(x)) \\
 St_\diamond(x, [R_i] \phi) &= \neg R_i(x, y) \vee St_\diamond(y, \phi), & & \text{where } y \text{ is a fresh variable.}
 \end{aligned}$$

The mapping  $St_\diamond$  is similar to the standard semantic translation  $St$  presented in [14]. It takes two arguments: a first-order term  $x$  that represents the world in which the modal formula is true, and the modal formula that has to be translated. The difference with the standard semantic translation is the translation of diamond formulae that in our case does not translate the semantic of the modal operator completely, and it has to be extended through other clauses.

Let us assume that the input set of the 3MG calculus contains  $m$  distinct diamond formulae of the form  $\langle R_i \rangle \phi'_j = \langle R_i \rangle (\phi'_{j_1} \wedge \dots \wedge \phi'_{j_k})$ . The new translation of the input set is the set  $N^{3MG}$  comprising of the following first-order clauses.

$$\begin{array}{ll}
 Q_{\phi_j}(u_i) & \text{for each } u_i : \phi_j \text{ in the input set} \\
 \neg Q_{\phi_j}(x) \vee St_\diamond(x, \phi_j) & \text{for each } u_i : \phi_j \text{ in the input set} \\
 \neg R_i(x, f_{\langle R_i \rangle \phi'_j}(x)) \vee St_\diamond(f_{\langle R_i \rangle \phi'_j}(x), \phi'_k) & \text{for each } 1 \leq j \leq m \\
 & \text{and each } j_1 \leq k \leq j_k \\
 R_i(u_j, v_k) & \text{for each } (u_j, v_k) : R_i \text{ in the input set}
 \end{array}$$

In case of reflexive or symmetric relations, the  $N^{3MG}$  set has to be extended with the following clauses.

$$\begin{array}{ll}
 \neg Q_{\phi_k}(x) \vee R_i(x, x) & \text{for each reflexive relation } R_i \\
 & \text{and each } u_j : \phi_k \text{ in the input set} \\
 \neg R_i(x, y) \vee R_j(x, x) & \text{for each reflexive relation } R_j \\
 \neg R_i(x, y) \vee R_j(y, y) & \text{for each reflexive relation } R_j \\
 \neg R_i(x, y) \vee R_i(y, x) & \text{for each symmetric relation } R_i
 \end{array}$$

The reflexivity property is usually encoded in first-order logic by the clause  $R_i(x, x)$ . This clause is not range-restricted and it is required by the PUHR approach that the input set of clauses is composed of only range-restricted clauses. Our alternative translation respects the range-restrictedness constraint and encodes the side conditions of the  $(\mathbf{T})^i$  rule in the 3MG calculus.

We refer to  $N^{3MG}$  as the *minimal Herbrand model preserving translation* of the input set.

For example, given the input set

$$\{w_1 : p_1, w_1 : \langle R_1 \rangle(p_2 \wedge \neg p_1), w_2 : ([R_1]p_2 \vee \langle R_1 \rangle(p_2 \wedge \neg p_1))\}$$

for  $\mathbf{K}_{(m)}$ , the translation  $N^{3MG}$  is the set of these clauses (where  $\phi'$  defines  $\langle R_1 \rangle p_2 \wedge \neg p_1$  and  $\phi''$  defines  $[R_1]p_2 \vee \langle R_1 \rangle(p_2 \wedge \neg p_1)$ ).

$$\begin{array}{ll}
 Q_{p_1}(w_1) & Q_{\phi'}(w_1) \\
 Q_{\phi''}(w_2) & \neg Q_{p_1}(x) \vee P_1(x) \\
 \neg Q_{\phi'}(x) \vee R_1(x, f_{\phi'}(x)) & \\
 \neg Q_{\phi''}(w_2) \vee \neg R_1(x, y) \vee P_2(y) \vee R_1(x, f_{\phi'}(x)) & \\
 \neg R_1(x, f_{\phi'}(x)) \vee P_2(f_{\phi'}(x)) & \neg R_1(x, f_{\phi'}(x)) \vee \neg P_1(f_{\phi'}(x))
 \end{array}$$

The translation to  $N^{3MG}$  can be seen to be the result of several modifications and transformations applied to the standard semantic translation of modal formulae into first-order logic. The definition of the standard semantic translation can be found in, e.g., [14] (where it is called the standard relational translation). We suppose the standard semantic translation of a tableau input is given by the semantic translation of each labelled formula  $u : \phi$  by using  $St(u, \phi)$ , and of each labelled relation  $(u, v) : R_i$  by translating it in  $R_i(u, v)$  (where  $St$  is defined as in [14]; see also [4, 18, 19]).

The first modification is the application of structural transformation on top of the semantic translation of the formulae in input. As we use structural transformation in a very restricted way, we omit its formal definition. Defini-

tions may be found in [16,4,18]. The reason of applying structural transformation, together with inner Skolemization, is to have the same Skolem functional symbol assigned to different occurrences of the same diamond formula. For example, given the following 3MG input

$$\{ u : [R_1](\langle R_1 \rangle p \vee q), v : \langle R_1 \rangle p, (u, v) : R_1 \}$$

the set of range-restricted clauses obtained by applying the semantic translation is as follows.

$$\begin{aligned} & \{ \neg R_1(u, x) \vee R(x, f_{\langle R_1 \rangle p}(x)) \vee Q(x), \\ & \neg R_1(u, x) \vee P(f_{\langle R_1 \rangle p}(x)) \vee Q(x), \\ & R_1(v, c_{\langle R_1 \rangle p}), P(c_{\langle R_1 \rangle p}), R_1(u, v) \} \end{aligned}$$

where the functional symbol  $f_{\langle R_1 \rangle p}$  and the constant symbol  $c_{\langle R_1 \rangle p}$  are the result of Skolemizing the two occurrences of  $\langle R_1 \rangle p$ . Due to these distinct Skolem symbols, an application of the PUHR approach to the set of clauses returns the two minimal Herbrand models

$$\begin{aligned} H_1 &= \{ Q(v), P(c_{\langle R_1 \rangle p}), R_1(u, v), R_1(v, c_{\langle R_1 \rangle p}) \} \\ H_2 &= \{ P(c_{\langle R_1 \rangle p}), P(f_{\langle R_1 \rangle p}(v)), R_1(u, v), R_1(v, c_{\langle R_1 \rangle p}), R_1(v, f_{\langle R_1 \rangle p}(v)) \} \end{aligned}$$

which are not minimal for the original modal input set. In fact, the 3MG calculus applied to the original set returns only the minimal modal Herbrand model

$$\{ f_{\langle R_1 \rangle p}(v) : p, (u, v) : R_1, (v, f_{\langle R_1 \rangle p}(v)) : R_1 \}.$$

To ensure that all Skolem functions introduced for the quantifiers that encode diamond formulae are unary (and not constants or have arity higher than one), structural transformation is used to introduce new symbols  $Q_{\phi_j}$  for the translation  $St(u_i, \phi_j)$  of all the labelled formulae in the input set of the 3MG calculus. We denote this transformation by  $\Xi$ . Usually structural transformation is used over a formula in which free variables occur ([18,4]). In the case of  $St(u, \phi)$  there are no free variables, but our use of the structural transformation can be justified by exploiting the equivalence  $St(u, \phi) \equiv \forall x(x \approx u \rightarrow St(u, \phi))$ . In addition, inner Skolemization is used to minimize the dependence in the Skolem terms on universally quantified variables. As a result there is a unique association between functional symbols and diamond subformulae. This also implies a one-to-one correspondence between the Herbrand universe of the first-order translation and the modal Herbrand universe of the original input.

**Lemma 4.2** *Let  $N$  be the input set of the 3MG calculus. Any set of tableau atoms  $I \models N$  iff  $H \models \Xi(N)$ , where  $H$  is a first-order Herbrand model representation of  $I$ .*

**Proof.** In [14] it is proved that given a modal formula  $\phi$ , for any modal model  $\mathcal{M}$  and world  $w$  in  $\mathcal{M}$  we have that  $\mathcal{M}, w \models \phi$  iff  $\mathcal{M} \models St(w, \phi)$ , where  $St$  is the standard semantic translation. In our case such theorem does not hold due to the different Skolem symbols assigned to different occurrences of the same diamond formula.

The use of structural transformation on top of the semantic translation solves the incongruity between the modal input and its translation into first-order logic, but it also adds new unary predicates that do not appear in the standard semantic translation. By construction the only possible ground instances of those new unary predicates appear in all possible Herbrand models. This implies that such predicates may be deleted from the model, resulting in a first-order representation of modal Herbrand models. Therefore, the elimination of those new predicates from the Herbrand models makes the theorem in [14] to hold for our definition of modal Herbrand model.  $\square$

As can be noted by our first-order encoding of the reflexivity property, this limited use of structural transformation allows us to reformulate the common clause for reflexivity in a range-restricted way without applying the range-restricted transformation described in [3], which would result in infinite Herbrand models and consequently the PUHR approach could not be successfully applied.

The second modification is a change to the translation of diamond formulae in  $St$ . Instead of  $St$  we use a mapping  $St'$  that is defined in the same way as  $St$  except the translation for diamond formulae is defined as follows:

$$St'(x, \langle R_i \rangle \phi) = \exists y (R_i(x, y) \wedge (\neg R_i(x, y) \vee St'(y, \phi))).$$

This new translation exploits the equivalence:

$$\exists y (R_i(x, y) \wedge (R_i(x, y) \rightarrow St'(y, \phi))) \equiv \exists y (R_i(x, y) \wedge St(y, \phi)).$$

This implies that the translation  $St'$  preserves logical equivalence. Due to the properties of the PUHR approach, using this modification of the translation we force the calculus to derive  $R_i(t, f_{\langle R_i \rangle}(t))$  before  $St'(f_{\langle R_i \rangle}(t), \phi)$  for any diamond formula  $\langle R_i \rangle \phi$ .

Let  $N$  be the set of first-order clauses obtained by translating the input of the 3MG calculus applying  $\Xi$  using the mapping  $St'$  to each labelled formula, and, if required, adding the clauses representing reflexivity and symmetry.

The next two lemmas justify the final most crucial modification of the translation.

**Lemma 4.3** *Let  $\mathcal{B}$  be a fully expanded, open branch in a tableau derivation generated by the PUHR approach applied to  $N$ . Suppose  $R_i(t, f_{\langle R_i \rangle}(t))$ , where  $t$  is a ground term occurring in a clause, is in  $\mathcal{B}$ . Then, the minimal*

Herbrand model  $H$  associated with  $\mathcal{B}$  is such that  $H \models St'(f_{\langle R_i \rangle \phi}(t), \phi)$ .

**Proof.** From the translation, only inference with clauses associated either with the translation of a diamond subformula, or with the symmetry clause can produce positive  $R_i$ -literals of the form  $R_i(t, f_{\langle R_i \rangle \phi}(t))$ . However, the symmetry clause infers  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  iff  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is already in the branch. That is, to infer  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  the symmetric relation  $R_i(f_{\langle R_i \rangle \phi}(t), t)$  must already be in the branch, and  $R_i(f_{\langle R_i \rangle \phi}(t), t)$  can be inferred only by the symmetric clause if  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is in the branch. For this reason we proceed the proof as if only the translation of a diamond subformula involves positive  $R_i$ -literals of the form  $R_i(t, f_{\langle R_i \rangle \phi}(t))$ .

Suppose  $\langle R_i \rangle \phi$  is an arbitrary occurrence of a diamond subformula of the 3MG input. Suppose the following are all the clauses in  $N$  associated with that occurrence of  $\langle R_i \rangle \phi$ :

$$\begin{aligned}
 & C \vee C' \vee R_i(x, f_{\langle R_i \rangle \phi}(x)) \\
 & C \vee C' \vee \neg R_i(x, f_{\langle R_i \rangle \phi}(x)) \vee T_1(f_{\langle R_i \rangle \phi}(x)) \\
 & \quad \vdots \\
 & C \vee C' \vee \neg R_i(x, f_{\langle R_i \rangle \phi}(x)) \vee T_k(f_{\langle R_i \rangle \phi}(x))
 \end{aligned} \tag{4}$$

where  $f_{\langle R_i \rangle \phi}$  is the Skolem term uniquely associated with  $\langle R_i \rangle \phi$ ,  $C = \neg A_1 \vee \dots \vee \neg A_n$ ,  $C' = B_1 \vee \dots \vee B_m$ ,  $T_1(f_{\langle R_i \rangle \phi}(x)) \wedge \dots \wedge T_k(f_{\langle R_i \rangle \phi}(x)) \equiv St'(f_{\langle R_i \rangle \phi}(x), \phi)$  and  $n \geq 0$ ,  $m \geq 0$ ,  $k \geq 0$ . Note  $C'$  is a positive clause, and the only other place where positive literals can occur is in  $T_1(f_{\langle R_i \rangle \phi}(x)), \dots, T_k(f_{\langle R_i \rangle \phi}(x))$ .

This implies that  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is on the branch  $\mathcal{B}$  only if it was derived by applying the PUHR rule to the first clause in (4). This gives  $(C' \vee R_i(x, f_{\langle R_i \rangle \phi}(x)))\sigma$  where  $\sigma$  is ground and instantiates  $x$  with  $t$ . Suppose  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  was derived in this way. We consider two cases.

Case  $m = 0$ , that is  $C'$  is empty. Then  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is produced by just applying the PUHR rule to the first clause in (4). As the PUHR approach is minimal model sound and complete, the model  $H$  extracted from  $\mathcal{B}$  is a minimal Herbrand model of  $N$ . We have that  $H \not\models C\sigma$ , because  $C\sigma$  is resolved by the PUHR rule, which means that the side premises make it false in  $H$ .  $H \not\models \neg R_i(t, f_{\langle R_i \rangle \phi}(t))$ , because  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is in the branch. This implies  $H \models T_1(f_{\langle R_i \rangle \phi}(t)), \dots, H \models T_k(f_{\langle R_i \rangle \phi}(t))$ . Hence,  $H \models T_1(f_{\langle R_i \rangle \phi}(t)) \wedge \dots \wedge T_k(f_{\langle R_i \rangle \phi}(t))$ . Therefore,  $H \models St'(f_{\langle R_i \rangle \phi}(t), \phi)$ .

Case  $m > 0$ , that is,  $C'$  is not empty. Then  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is the conclusion of applying the complement splitting rule. Suppose  $\neg B_{i_1}\sigma, \dots, \neg B_{i_l}\sigma$ , where  $1 \leq l \leq m$ , are the other ground literals in  $\mathcal{B}$  produced by this application of the complement splitting rule. We consider two subcases in which

we focus on any of the clauses of the form:

$$C \vee C' \vee \neg R_i(x, f_{\langle R_i \rangle \phi}(x)) \vee T_i(f_{\langle R_i \rangle \phi}(x)). \quad (5)$$

(a) The PUHR rule is not applicable to (5). This means that  $T_i(f_{\langle R_i \rangle \phi}(x))$  contains at least a negative literal  $L$  and its complement is not in  $\mathcal{B}$ . By construction,  $H$  is such that  $H \not\models C\sigma$ ,  $H \not\models \neg R_i(t, f_{\langle R_i \rangle \phi}(t))$  but (5) is true in  $H$ . This implies,  $H \models C'\sigma \vee T_i(f_{\langle R_i \rangle \phi}(t))$ . Since the PUHR rule is not applicable to (5),  $H \models L\sigma$ . Therefore,  $H \models T_i(f_{\langle R_i \rangle \phi}(t))$ .

(b) The PUHR is applicable to (5). This means that  $T_i(f_{\langle R_i \rangle \phi}(x))$  either contains only positive literals (b1), or for all its negative literals there are corresponding premises in the application of the PUHR rule (b2).

(b1) The clause derived by the application of the PUHR rule to (5) is  $C''\sigma \vee T_i(f_{\langle R_i \rangle \phi}(t))$ , where  $C''$  is the clause  $C'$  but all the literals  $B_{i_j}$  are removed ( $1 \leq j \leq l$ ). If one of the literals in  $C''\sigma$  is true in  $H$  then, since  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is also true in  $H$ ,  $H$  is not a minimal model, because each of the clauses in (4) are true in  $H \setminus \{R_i(t, f_{\langle R_i \rangle \phi}(t))\}$ . This is a contradiction. Thus, one of the literals in  $T_i(f_{\langle R_i \rangle \phi}(t))$  is true in  $H$ . Therefore,  $H \models T_i(f_{\langle R_i \rangle \phi}(t))$ .

(b2) The clause derived by the application of the PUHR rule to (5) is  $C''\sigma \vee T'_i(f_{\langle R_i \rangle \phi'}(t))$ , where  $C''\sigma$  is as in (b1) and  $T'_i(f_{\langle R_i \rangle \phi'}(t)) \subseteq T_i(f_{\langle R_i \rangle \phi'}(t))$ . Specifically,  $T'_i(f_{\langle R_i \rangle \phi'}(t))$  is as  $T_i(f_{\langle R_i \rangle \phi'}(x))\sigma$  without all the negative literals in  $T_i(f_{\langle R_i \rangle \phi'}(x))$ . Suppose  $T'_i(f_{\langle R_i \rangle \phi'}(t))$  is empty. This means that  $H \models C''\sigma$ . But this leads to the same contradiction of (b1). Since  $T'_i(f_{\langle R_i \rangle \phi'}(t))$  is empty,  $H$  does not satisfy  $T_i(f_{\langle R_i \rangle \phi'}(t))$ . So,  $H$  does not satisfy (5) and the branch is closed. This is a contradiction because the branch is supposed to be open. Suppose  $T'_i(f_{\langle R_i \rangle \phi'}(t))$  is not empty. For the same reasoning as for (b1), the model  $H$  cannot satisfy  $C''\sigma$ . Thus,  $H \models T'_i(f_{\langle R_i \rangle \phi'}(t))$ . Therefore,  $H \models T_i(f_{\langle R_i \rangle \phi'}(t))$ .  $\square$

**Lemma 4.4** *Let  $N$  be a set of clauses as in Lemma 4.3. Then, each model  $H$  extracted from a branch of a tableau generated by the PUHR approach applied to  $N$  is such that*

$$H \models St'(f_{\langle R_i \rangle \phi}(t), \phi) \text{ iff } H \models R_i(t, f_{\langle R_i \rangle \phi}(t)).$$

**Proof.** The new translation for diamond formulae in  $St'$  ensures that a minimal Herbrand model  $H$  satisfies  $St'(f_{\langle R_i \rangle \phi}(t), \phi)$  only if  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is in  $H$ . That is because the new translation is such that  $\neg R_i(x, f_{\langle R_i \rangle \phi}(x))$  appears in front of each  $T_i(f_{\langle R_i \rangle \phi}(x))$ . Hence,  $T_i(f_{\langle R_i \rangle \phi}(t))$  is on the branch only if  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  already is. Lemma 4.3 asserts that if  $R_i(t, f_{\langle R_i \rangle \phi}(t))$  is in  $H$  then  $H \models St'(f_{\langle R_i \rangle \phi}(t), \phi)$ . Therefore, a minimal model  $H$  is such that  $H \models St'(f_{\langle R_i \rangle \phi}(t), \phi)$  iff  $H \models R_i(t, f_{\langle R_i \rangle \phi}(t))$ .  $\square$

Lemmas 4.3 and 4.4 justify the most crucial step in the modification to the translation. Let  $F_0$  be the result of applying  $\Xi$  using the mapping  $St'$  to an arbitrary labelled formula of the 3MG input. Suppose that the labelled formula contains  $n$  distinct diamond subformulae  $\langle R_i \rangle \phi_1, \dots, \langle R_n \rangle \phi_n$ . We iteratively apply the following transformation for each  $1 \leq i \leq n$ :

$$F_i = F_{i-1}[St'(x, \langle R_i \rangle \phi_i) / R_i(x, f_{\langle R_i \rangle \phi_i}(x))] \\ \wedge \forall x (\neg R_i(x, f_{\langle R_i \rangle \phi_i}(x)) \vee St'(f_{\langle R_i \rangle \phi_i}(x), \phi_i)).$$

The resulting formula  $F_n$  obtained in this way is the *diamond optimization transformation* of  $F_0$ . Intuitively, diamond optimisation transformation is similar to structural transformation but instead of introducing a new symbol it reuses the literal  $R_i(x, f_{\langle R_i \rangle \phi_i}(x))$  for the translation of a diamond subformula. The formula  $\forall x (\neg R_i(x, f_{\langle R_i \rangle \phi_i}(x)) \vee St'(f_{\langle R_i \rangle \phi_i}(x), \phi_i))$  can be seen as the (Skolemized) definition of  $R_i(x, f_{\langle R_i \rangle \phi_i}(x))$  and  $St'(x, \langle R_i \rangle \phi_i)$ .

The diamond optimization transformation is the last step of our specialized translation. The  $N^{3MG}$  set is the final result of applying our translation to the input set of the 3MG calculus.

The last step before showing the bijection between the 3MG calculus and the PUHR approach is to prove that the main requirement is respected: the minimal modal Herbrand models of any input to the 3MG calculus correspond to the minimal Herbrand models of the new translation of it. The correspondence is such that every minimal modal Herbrand model may be translated to a minimal Herbrand model and vice versa. The translation is as usual:  $u : p_i \Rightarrow P_i(u)$ ,  $(u, v) : R_i \Rightarrow R_i(u, v)$ , and vice versa. This correspondence exists iff all atoms  $Q_{\phi_j}(u_i)$  have been filtered from the Herbrand models.

**Lemma 4.5** *Suppose  $N$  is the input of the 3MG calculus, and  $N^{3MG}$  is the result of the minimal Herbrand model preserving translation. The set of minimal modal Herbrand models of  $N$  corresponds to the set of minimal Herbrand models of  $N^{3MG}$ .*

**Proof.** Lemma 4.2 establishes that to each Herbrand model of  $N^{3MG}$  correspond a modal Herbrand model of  $N$ , and the other way around. This holds also after applying the other two steps of the transformation because both of them preserve logical equivalence.

Let us assume that there exists a minimal Herbrand model  $H$  of  $N^{3MG}$  that does not represent a minimal modal Herbrand model  $I$  of  $N$ . This implies, that there is a minimal modal Herbrand model  $I'$  which is minimal with respect to  $I$ . Then, by Lemma 4.2 there is an Herbrand model  $H'$  of  $N$  corresponding to  $I'$ . This leads to a contradiction because  $H' \subset H$ , and  $H$  cannot be minimal. This implies, that to each minimal Herbrand model of  $N^{3MG}$  corresponds a

minimal modal Herbrand model of  $N$ .

Following an analogous reasoning, it is possible to prove that to each minimal modal Herbrand model of  $N$  corresponds a minimal Herbrand model of  $N^{3MG}$ . Therefore, the set of minimal modal Herbrand models of  $N$  corresponds to the set of minimal Herbrand models of  $N^{3MG}$ .  $\square$

### 4.3 Minimal Model Soundness and Completeness

For the minimal model soundness and completeness of the 3MG calculus, what remains to be shown is a correspondence between 3MG derivations and PUHR derivations. We do this by defining a variation of the PUHR approach, called the *SBR approach*, and showing: (i) there is a bisimulation relationship preserving minimal Herbrand models between the PUHR approach and the SBR approach, and (ii) there is a bisimulation relationship preserving minimal Herbrand models between the SBR approach and the 3MG calculus.

By the *SBR approach* we mean the variation of the PUHR approach where the PUHR rule is replaced by a selection-based resolution rule, namely:

$$\text{SBR rule: } \frac{A_1 \quad \dots \quad A_j \quad \neg A'_1 \vee \dots \vee \neg A'_n \vee D}{(\neg A'_{j+1} \vee \dots \vee \neg A'_n \vee D)\sigma}$$

where (i)  $A_1, \dots, A_j$  are positive ground unit clauses (atoms), (ii)  $\neg A'_1, \dots, \neg A'_j$  in the main premise are selected by a selection function  $S$  (for  $1 \leq j \leq n$ ), (iii)  $D$  does not contain any negative literals, and (iv)  $A_1 = A'_1\sigma, \dots, A_j = A'_j\sigma$ . This means the selection function  $S$  always selects at least one negative literal from each clause containing negative literals.

**Theorem 4.6** *Let  $N$  be a finite set of range-restricted clauses. If  $N$  has finitely many finite Herbrand models, then the SBR approach applied to  $N$  terminates, it returns all and only minimal Herbrand models of  $N$ , and does not return any minimal model more than once.*

**Proof.** The PUHR rule is a hyperresolution rule and the SBR rule is a selection-based resolution rule. It is well-known that the hyperresolution rule can be simulated by a bounded number of selection-based resolution steps. This implies that the PUHR approach is step-wise simulated by the SBR approach and is a specialization of it. All that needs to be proved is that the minimal Herbrand models generated by the two approaches on the same input are the same. Unless all negative literals are selected in a clause, applications of the SBR rule lead to non-positive clauses that cannot be split to contribute to the model. However, it is not difficult to prove that repeated applications of the SBR rule lead to a positive ground clause  $C$  iff one application of the

PUHR rule produces  $C$ . Therefore, the SBR rule can substitute the PUHR rule and this substitution preserves the minimal model soundness and completeness.  $\square$

The selection functions can be defined in an arbitrary way as long as it selects only negative literals. For our purposes, we define  $S$  as follows:

$$S(C) = \begin{cases} \{\neg Q_{\phi_j}(x)\} & \text{if } C = \neg Q_{\phi_j}(x) \vee D \\ \{\neg R_i(x, y)\} & \text{if } C = \neg R_i(x, y) \vee D \\ \{\neg A_1, \dots, \neg A_n\} & \text{if } C = \neg A_1 \vee \dots \vee \neg A_n \vee D. \end{cases} \quad (6)$$

In the third case each  $\neg A_i$  denotes a ground unary literal or a binary literal with at least one ground argument, and  $D$  is a clause which may contain both positive and negative literals.

The selection of  $\neg Q_{\phi_j}(x)$  in clauses of the form  $\neg Q_{\phi_j}(x) \vee D$  is used to obtain clauses that correspond to the input of the 3MG calculus and to the application of reflexivity to the terms that tag labelled formulae in the 3MG input.

The selection of  $\neg R_i(x, y)$  in clauses of the form  $\neg R_i(x, y) \vee D$  is a consequence of the diamond optimization transformation, the representation of reflexivity and the presence of symmetry clauses. It is used to simulate the  $(\diamond)$  rule, the  $(\mathbf{T})^i$  rule and the  $(\mathbf{B})^i$  rule.

The selection of  $\neg A_1, \dots, \neg A_n$  in the remaining cases simulates the  $(SBR)$  rule of the 3MG calculus. Its aim is to select the translation of all labelled negative atoms. Their representation in first-order logic is as follows:  $u : \neg p$  is represented by  $\neg P(u)$ ,  $u : \Box\phi$  is represented by  $\neg R_i(u, x) \vee St_{\diamond}(x, \phi)$ , and  $(u, v) : \neg R$  is represented by  $\neg R_i(u, v)$ , where  $u$  and  $v$  are ground terms.

Because of the way the minimal Herbrand model preserving translation is defined, the defined selection function  $S$  respects the prerequisite of the SBR rule:  $S$  selects at least a negative literal from each clause containing negative literals. This means that Theorem 4.6 holds for the SBR approach using the selection function  $S$ . Subsequently, when we refer to the SBR approach we mean the instance where  $S$  is as defined in (6) above.

Next, we show that there exists a bijection between the 3MG calculus and the SBR approach applied to the minimal Herbrand model preserving translation.

**Lemma 4.7** *Let  $N$  be the input to the 3MG calculus. Suppose  $N^{3MG}$  is the set of first-order clauses obtained by applying the minimal Herbrand model preserving translation to  $N$ . Suppose the 3MG calculus is applied to  $N$ , and the SBR approach is applied to  $N^{3MG}$ . Then there exists a bijection between:*

- (i) *an application of the  $(\diamond)$  rule and a bounded number of repeated applications of the SBR rule with main premise  $\neg R_i(x, f_{\langle R_i \rangle \phi_i}(x)) \vee D$ ,*

- (ii) *an application of the (CS) rule and an application of the complement splitting rule,*
- (iii) *an application of the (SBR) rule and an application of the SBR rule with main premise  $\neg A_1 \vee \dots \vee \neg A_n \vee D$ ,*
- (iv) *an application of the  $(\mathbf{T})^i$  rule, where the term belongs to a tableau formula in  $N$  of the form  $u : \phi$ , and application of the SBR rule with main premise  $\neg Q_\phi(x) \vee D$ ,*
- (v) *an application of the  $(\mathbf{T})^i$  rule, where the term belongs to a tableau formula of the form  $(u, v) : R_j$ , and application of the SBR rule with main premise  $\neg R_j(x, y) \vee D$ , and*
- (vi) *an application of the  $(\mathbf{B})^i$  rule and application of the SBR rule with main premise  $\neg R_i(x, y) \vee D$ , and*

**Proof.** (Sketch): The simulation required the use of a mapping function from the modal tableau to the first-order tableau, and a mapping for the reverse direction. The mapping used for the simulations are based on a generalization of the new translation from modal formulae into first-order sets of clauses and its inverse.

(i) The  $(\diamond)$  rule can be seen as the union of the standard diamond rule and an  $\alpha$  rule of the form  $u : \alpha_1 \wedge \dots \wedge \alpha_n / u : \alpha_i$ , for each  $1 \leq i \leq n$ . The standard diamond rule creates a labelled relation and a modal formula expandable via an  $\alpha$  rule. The labelled relation is the side premise necessary for the application of the SBR rule in the SBR approach. Each application of an  $\alpha$  rule corresponds to an application of the SBR rule where  $S$  selects  $\neg R(x, f_\diamond(x))$  and vice versa. Specifically, each application of an  $\alpha$  rule adds  $u : \alpha_i$  which represents exactly the result of the application of the SBR rule:  $St_\diamond(u, \alpha_i)$ . The reasoning is valid also for the other way around.

In the case of (ii), (iii), (iv), (v) and (vi) there exists a one-to-one correspondence between 3MG steps and steps in the SBR approach.  $\square$

**Lemma 4.8** *Let  $N$  be the input of the 3MG calculus. There exists a bisimulation between the 3MG calculus applied to  $N$  and the SBR approach applied to the minimal Herbrand model preserving translation of  $N$ .*

**Proof.** By Lemma 4.7 there are several bijections between rules of the 3MG calculus and the rules of the SBR approach. Two cases not considered in Lemma 4.7 are the  $(\vee)_E$  rule and the SBR rule (of the SBR approach) where the main premise is  $\neg Q_{\phi_j}(x) \vee D$  and  $D$  is not an R-literal. The  $(\vee)_E$  has no correspondent in the SBR approach. It just distributes the  $:$  operator over disjunction based on a logical equivalence and does not contribute to the model. This means the  $(\vee)_E$  rule preserves minimal model soundness and completeness.

In the other case,  $Q_{\phi_j}(u_i)$  encodes the labelled formulae that belongs to  $N$

and applying a resolution step to  $\neg Q_{\phi_j}(x) \vee D$  allows us to have a first-order representation of the modal input. This means that the SBR rule where  $S$  selects  $\neg Q_{\phi_j}(x)$  when  $D$  is not an R-literal does have a real correspondent in the 3MG calculus.

These characteristic of the  $(\vee)_E$  rule and the SBR rule selecting  $\neg Q_{\phi_j}$  when  $D$  is not an R-literal, and the result of Lemma 4.7 imply the existence of a bisimulation between the 3MG calculus applied to  $N$  and the SBR approach applied to the minimal Herbrand model preserving translation of  $N$ .  $\square$

**Theorem 4.9 (Minimal Model Soundness and Completeness)** *Let  $N$  be a set of tableau clauses. The 3MG calculus applied to  $N$  terminates, generates all and only minimal modal Herbrand model of  $N$ , and each model is generated no more than once.*

**Proof.** By Lemma 4.8, the 3MG calculus is equivalent to the SBR approach. By Theorem 4.6, the SBR approach and the PUHR approach have the same behaviour on the same input and produce the same minimal Herbrand models. Hence, by Lemma 4.5 and Theorem 4.6, the 3MG calculus is minimal model sound and complete.

Complement splitting ensures that a model is generated only once. This can be inferred by the correspondence between models generated by the 3MG calculus and the SBR approach, for which this characteristic has been proved.

For termination we need to show that in any branch there cannot be an infinite sequence of steps. This can be done by defining a measure on tableau clauses and showing that the measure strictly decreases with every inference step. This can be done similar to [4] (Theorem 7.7 on p. 281), where termination is proved for a specific resolution calculus applied to an appropriate translation of formulae belonging to a logic in between  $\mathbf{K}$  and  $\mathbf{K}_{(m)}(\cap, \cup, \smile)$  into first-order clauses.  $\square$

## 5 Discussion

The presented 3MG calculus generates all and only minimal modal Herbrand models working directly with modal formulae. To our knowledge there are no other methods having the same aim and working directly with modal formulae.

The closest approach that aims to generate all and only minimal Herbrand models for modal logic is described in [5]. The approach in [5] covers the  $\text{GF}^-$  fragment of first-order logic that is wider than the multi-modal logic  $\mathbf{K}_{(m)}$  and extensions with axioms  $\mathbf{T}$  and  $\mathbf{B}$ . However, the approach does not operate on modal formulae but on their translation into first-order logic. Its results differ slightly from the results obtained using the 3MG calculus. Specifically, the set of models generated by our calculus is a subset of the set of models obtained using the method of [5]. This is due to the particular structural

transformation used during the translation from modal formulae into first-order formulae. For instance, with the 3MG calculus the only minimal modal Herbrand model of the tableau clause  $\Delta = w : (p_1 \vee [R_1]p_2)$  is the empty set. If we translate  $\Delta$  into a first-order formula introducing a new symbol corresponding to  $[R_1]p_2$ , then in the resulting set of clauses there is a positive clause of the form  $P_1(w) \vee Q_{[R_1]p_2}(w)$ , which may produce a model not produced by the 3MG calculus. This implies it is possible to create models that are not minimal, at least not under our definition of minimal modal Herbrand model, for the original modal formula, but are minimal for the first-order clauses.

Even though there is some similarity between some of the rules, the 3MG calculus and the PUHR calculus [3] do not correspond directly to each other in the sense that a step in the modal calculus can be simulated by one or more steps of the calculus of the PUHR approach, or the other way around. As said in Section 4, and explained in detail in [15], we prove minimal model soundness and completeness of the 3MG calculus by showing that the two calculi are approximations of each other via a non-trivial new translation of modal formulae to first-order clause form. Our new translation has the additional benefit that it leads to the reduction of the search space in the application of the PUHR approach to clauses obtained by the standard translation.

The use of the (*SBR*) rule in the 3MG calculus represents another important difference to the PUHR approach. It is possible to use a modal version of the PUHR rule where the box formulae are expanded away into disjunctions of tableau literals instead of the (*SBR*) rule. The benefits would be fewer inference steps and no intermediary clauses would be produced. It is not clear whether this would improve efficiency significantly though. A successful application of a multiple premise rule like the PUHR rule requires combinatorially many matching attempts to find the right premises for performing an inference step. Thus, we do not expect there to be significant degrading in performance through the use of the (*SBR*) rule. As the (*SBR*) rule can be viewed as the composition of the standard closure rule and expansion rules for multi-modal tableaux calculi, it has the advantage that it is more ‘modal’, and thus more natural, than a modal tableau version of the PUHR rule, which would require the introduction of more notation non-standard for modal tableau calculi.

In [7] the authors present a tableau calculus for circumscriptive reasoning for the description logic  $\mathcal{ALCO}$ . Circumscription aims to minimize a specific set of predicates. This means that our calculus can be thought as the particular circumscription case in which all predicates are minimized. However, the calculus presented in [7] differs from the 3MG calculus in aim, the logic considered, and the methodology used. First, the calculus in [7] does not actually generate minimal models (because  $R$  predicates cannot be minimised). Its task is to check if a formula is entailed by an interpretation respecting a circumscriptive pattern. Second, their tableau calculus is for the description

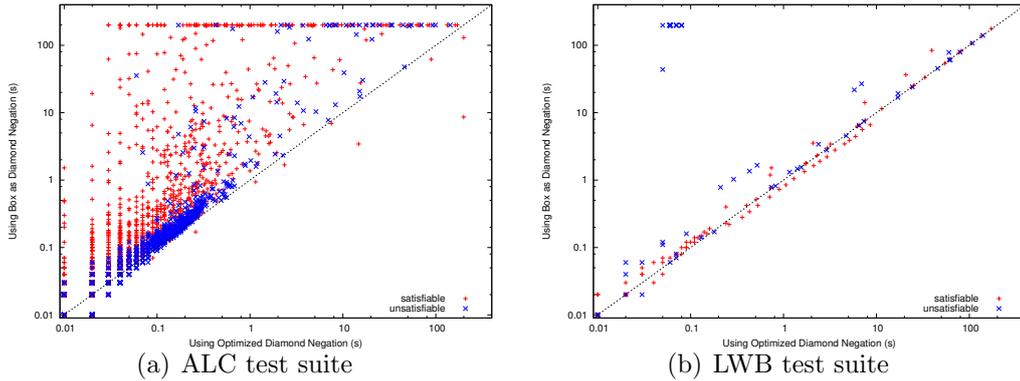


Fig. 3. Comparison of CPU runtimes in seconds of using the *neg* function

logic  $\mathcal{ALCO}$  which has different expressive power compared to multi-modal logic  $\mathbf{K}_{(m)}$  extended with reflexivity and symmetry. Finally, their method involves the use of intermediate tests during the tableau derivation, a technique also used in [12] that presents a tableau calculus for circumscription in function-free first-order clauses. The 3MG calculus does not require any separate test, because minimality is ensured by the (*CS*) rule and model constraint propagation.

The complement splitting rule (*CS*) in the 3MG calculus is based on the *neg* function rather than standard negation, because standard negation of diamond formulae can lead to non-minimal models. For example, given a positive clause  $w : p_1 \vee w : \langle R_1 \rangle \neg p_2$ , the negation of  $w : \langle R_1 \rangle \neg p_2$  is  $w : [R_1] p_2$ . The expansion of the box formula could result in a non-minimal model. The problem is that a model created on this branch might include positive literals originating from below the box operator. This could cause a minimal model to be missed. For this reason the *neg* function defines the negation of a labelled diamond formula as the negation of the relation that this diamond would create.

With the view of using this technique for other related logics, it is important to note that it is not possible to use the *neg* function if the negated labelled relation resulting from it can be generated by rules other than the ( $\diamond$ ) rule.

The use of a complement splitting rule in tableau calculi is not new, though typically the rule would be used in the form  $u : \phi_1 \vee \phi_2 / u : \phi_1 \mid u : \phi_2, u : \neg \phi_1$ . This ensures that branches are disjoint and avoids repeated inference steps, but can lead to significant degrading of performance due to the additional expansion of  $u : \neg \phi_1$ . To test whether our form of complement splitting based on the *neg* function can be an effective way of achieving a reduction in search space without degrading the performance, we have implemented a simple tableau prover for  $\mathbf{K}_{(m)}$ . The main optimization in the implementation is

the use of backjumping, a clever form of backtracking. The different diamond negation is obtained by setting a flag. We decided to implement complement splitting only for diamond formulae to avoid inferences due to the negation of other complex formulae like box formulae or conjunctions. Tests were run with a timeout of 200 seconds on a Dell machine with Intel Core2 Duo Processor E6300 (2M Cache, 1.86 GHz), and 1GB of main memory. Figure 3 shows the CPU time comparisons with and without our optimized diamond negation over the ALC test suite from <http://www.csc.liv.ac.uk/~ullrich/ALC1996/> (collection of problems in alc-3-1-x-3-2 and alc-3-1-x-3-5), and the LWB test suite from <http://iamwww.unibe.ch/~lwb/benchmarks/benchmarks.html>. That there are more points above the diagonal of both graphs indicates superior performance for complement splitting with optimized diamond negation for both test suites. For the LWB test suite the advantage is however less prominent. This is perhaps not surprising because the problems in the ALC test suite are known to contain more redundancies. It is interesting to note that almost all the cases in which the standard negation works better in the LWB test suite are for the class  $k\_grz\_n$ , while for all other classes the result is either in favour of our optimized negation or there is no evident difference.

## 6 Conclusion

We introduced the notion of (minimal) modal Herbrand model and defined a multi-modal tableau calculus which generates all and only minimal modal Herbrand models. We proved minimal model soundness and completeness by devising a new translation from modal formulae into first-order clause sets, and showing that there exists a correspondence between the 3MG calculus and a variation of the PUHR approach based on selection-based resolution. A positive side-effect of the function  $neg$  for the negation of diamond formulae is a reduction of the search space for the 3MG approach as the experiments have shown (we believe these can be carried over to the PUHR approach, as we intend to show in a future work).

A possible extension of our approach includes lazy clausification performed by appropriate extension rules. This would not improve the theoretical worst-case complexity of the calculus but will be of practical benefit in a future implementation. The calculus presented in this paper does not use lazy clausification because it makes it easier to prove its soundness and completeness via the connection with the PUHR approach, which is a purely clausal approach.

This work is just a starting point of developing more general tableau calculi for minimal model generation. Extensions in several directions are possible. In this paper we focused only on minimal model generation of multi-modal logic  $\mathbf{K}_{(m)}$  and its extension with axioms  $\mathbf{T}$  and  $\mathbf{B}$ . An interesting extension is

the addition of other well-known axioms such as **D**, **4** and **5** that correspond to seriality, transitivity and euclideaness of the accessibility relations. The main challenge is to understand how to deal with the introduction of a blocking technique, which would avoid the possibility of generating infinite models without compromising soundness and completeness of the calculus. The minimal model generation for dynamic modal logics [19] and description logics, such as *ALCO* and *ALCOIQ* (for which circumscription has been studied in [2,7]), is another possible direction for future work.

## References

- [1] P. Baumgartner, P. Fröhlich, U. Furbach, and W. Nejdl. Tableaux for diagnosis applications. In *Proc. TABLEAUX'97*, volume 1227 of *LNCS*, pages 76–90. Springer, 1997.
- [2] P. A. Bonatti, C. Lutz, and F. Wolter. The complexity of circumscription in description logic. *J. Artif. Int. Res.*, 35:717–773, 2009.
- [3] F. Bry and A. Yahya. Positive unit hyperresolution tableaux and their application to minimal model generation. *J. Automated Reasoning*, 25(1):35–82, 2000.
- [4] H. De Nivelle, R. A. Schmidt, and U. Hustadt. Resolution-based methods for modal logics. *Logic J. IGPL*, 8(3):265–292, 2000.
- [5] L. Georgieva, U. Hustadt, and R. A. Schmidt. Computational space efficiency and minimal model generation for guarded formulae. In *Proc. LPAR'01*, volume 2250 of *LNAI*. Springer, 2001.
- [6] L. Giordano, V. Gliozzi, N. Olivetti, and G. L. Pozzato. Reasoning about typicality in preferential description logics. In *Proc. JELIA'08*, volume 5293 of *LNCS*, pages 192–205. Springer, 2008.
- [7] S. Grimm and P. Hitzler. A preferential tableaux calculus for circumscriptive *ALCO*. In *Proc. RR'09*, volume 5837 of *LNCS*, pages 40–54. Springer, 2009.
- [8] J. Hintikka. Model minimization - an alternative to circumscription. *J. Automated Reasoning*, 4(1):1–13, 1988.
- [9] U. Hustadt and R. A. Schmidt. Using resolution for testing modal satisfiability and building models. *J. Automated Reasoning*, 28(2):205–232, February 2002.
- [10] S. Lorenz. A tableaux prover for domain minimization. *J. Automated Reasoning*, 13(3):375–390, 1994.
- [11] J. McCarthy. Circumscription - a form of non-monotonic reasoning. *Artificial Intelligence*, 13:27–39, 1980.
- [12] I. Niemelä. Implementing circumscription using a tableau method. In *Proc. ECAI '96*, pages 80–84. Wiley, 1996.
- [13] I. Niemelä. A tableau calculus for minimal model reasoning. In *Proc. TABLEAUX'96*, volume 1071 of *LNCS*, pages 278–294. Springer, 1996.
- [14] H. J. Ohlbach, A. Nonnengart, M. de Rijke, and D. M. Gabbay. Encoding two-valued nonclassical logics in classical logic. In J. A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, pages 1403–1486. Elsevier, 2001.
- [15] F. Papacchini and R. A. Schmidt. A tableau calculus for minimal modal model generation. Preprint, School of Computer Science, University of Manchester. <http://www.cs.man.ac.uk/~papacchf/PapacchiniSchmidt11a.pdf>.
- [16] D. A. Plaisted and S. Greenbaum. A structure-preserving clause form translation. *J. Symb. Comput.*, 2(3):293–304, 1986.

- [17] R. Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 32(1):57–95, 1987.
- [18] R. A. Schmidt and U. Hustadt. Mechanised reasoning and model generation for extended modal logics. *LNCIS*, 2929:38–67, 2003.
- [19] R. A. Schmidt and U. Hustadt. First-order resolution methods for modal logics, 2006. Manuscript. To appear in *Volume in memoriam of Harald Ganzinger*. <http://www.cs.man.ac.uk/~schmidt/publications/SchmidtHustadt06a.html>.