A Modal-Layered Resolution Calculus for K

Cláudia Nalon[1], Ullrich Hustadt[2], and Clare Dixon[2]

[1] Departament of Computer Science, University of Brasília
C.P. 4466 – CEP:70.910-090 – Brasília – DF – Brazil
nalon@unb.br

[2] Department of Computer Science, University of Liverpool
Liverpool, L69 3BX – United Kingdom
U.Hustadt, CLDixon@liverpool.ac.uk

Abstract. Resolution-based provers for multimodal normal logics require pruning of the search space for a proof in order to deal with the inherent intractability of the satisfiability problem for such logics. We present a clausal modal-layered hyper-resolution calculus for the basic multimodal logic, which divides the clause set according to the modal depth at which clauses occur. We show that the calculus is complete for the logics being considered. We also show that the calculus can be combined with other strategies. In particular, we discuss the completeness of combining modal layering and negative resolution. In addition, we present an incompleteness result for modal layering together with ordered resolution.

Keywords: Automated reasoning, normal modal logics, resolution method

1 Introduction

Automatic theorem-proving for the multimodal basic modal logic K_n has attracted the interest of researchers as this logic is able to express non-trivial problems in Artificial Intelligence and other areas. For instance, it is well-known that the description logic ALC, which has been applied to terminological representation, is a syntactic variant of K_n [20]. Problems in Quantified Boolean Propositional Logic, which is a very active area in the SAT community, can also be translated into K_n.

The reasoning tasks in K_n are far from trivial. Given a formula φ, the local satisfiability problem consists of showing that there is a world in a model that satisfies φ. A formula φ is globally satisfiable if there is a model such that all worlds in this model satisfy φ. Given a set of formulae Γ and a formula φ, the local satisfiability of φ under the global constraints Γ consists of showing that there is a model that globally satisfies the formulae in Γ and that there is a world in this model that satisfies φ. The local satisfiability problem for the multimodal case is PSPACE-complete [9]. The global satisfiability and the local satisfiability under global constraints problems for K_n are EXPTIME-complete [24].

Several proof methods and tools for reasoning in K_n exist, either in the form of methods applied direct to the modal language or obtained by translation into more expressive languages (First-Order Logic, for instance). Translation-based methods benefit not only from the existence of available theorem-provers, therefore not requiring big effort for implementation, but strategies available for the object language can be almost immediately applied to the translated problem [11]. This is not the case for direct methods, where strategies need to be adapted to deal with the underlying normal forms and inference rules. In this paper, we propose a resolution-based proof method for K_n and investigate the completeness of strategies for such method.

The calculus presented here borrows from previous work in several aspects. Firstly, it requires a translation into a more expressive modal language, where labels are used to express semantic properties of a formula. Secondly, it makes use of labelled resolution in order to avoid unnecessary applications of the inference rules. For instance, in the unrestricted resolution method for K_n [16], the translation of $\Diamond_a \Diamond_a p \wedge \Box_a \neg p$ into the normal form results in the set $\{\textbf{start} \Rightarrow t_0, t_0 \Rightarrow \Diamond_a t_1, t_1 \Rightarrow \Diamond_a p, t_0 \Rightarrow \Box_a \neg p\}$. The application of resolution to clauses $t_1 \Rightarrow \Diamond_a p$ and $t_0 \Rightarrow \Box_a \neg p$ is not desirable, as as $\Diamond_a p$ and $\Box_a \neg p$ occur at different modal levels and are not, in fact, contradictory. The translation into the normal form given here leads to the direct implementation of the layered modal heuristic given in [2]. However, in [2] the modal levels are hard-coded in the names of the translated propositional symbols, which requires the implementation of unification, making the application of both local and global reasoning more difficult. Besides, our approach might effectively lead to a good partition of the clause set, restricting the application of the inference rules to (possibly) smaller sets, which can improve the performance of reasoners [22].

In [1], a labelled non-clausal resolution-based proof method for ALC is given. Formulae are labelled by either constants a, which corresponds to names of worlds in a model, or by pairs (a,b) representing the relation between two worlds named by a and b, respectively. Our calculus is similar, but labels correspond to modal levels instead of worlds. Having worlds as labels might require repeated applications of global reasoning for worlds at the same modal level. Labelled resolution is also used in e.g. [8], where (sets of) labels express the semantic constraints in multi-valued logics. We have chosen to keep the labels simple as unification only requires a simple check. However, by extending the labels to sets, the calculus can be easily adapted to deal with the satisfiability problem for other interesting modal logics (e.g. graded modalities).

The paper is organised as follows. Section 2 presents the language of K_n. The normal form and the modal-layered based calculus are presented in Sections 3 and 4. Correctness is proved in Section 5. In Section 6, we show that the application of the calculus can be restricted to negative resolution. Ordering refinements, discussed in Section 7, are shown to be incomplete for the particular calculus given here.

2 Language

The set $\mathsf{WFF}_{\mathsf{K}_n}$ of *well-formed formulae* of the logic K_n is constructed from a denumerable set of *propositional symbols*, $P = \{p, q, p', q', p_1, q_1, \ldots\}$, the negation symbol $\neg$, the conjunction symbol $\wedge$, the propositional constant **true**, and the unary connective $\Box_a$ for each index a in a finite, fixed set $A_n = \{1, \ldots, n\}$, $n \in \mathbb{N}$.

Definition 1. *The* set of well-formed formulae*,* $\mathsf{WFF}_{\mathsf{K}_n}$*, is the least set such that* $p \in P$ *and* **true** *are in* $\mathsf{WFF}_{\mathsf{K}_n}$*; if* φ *and* ψ *are in* $\mathsf{WFF}_{\mathsf{K}_n}$*, then so are* $\neg\varphi$*,* $(\varphi \wedge \psi)$*, and* $\Box_a \varphi$ *for each* $a \in A_n$*.*

When $n = 1$, we often omit the index, that is, $\Box\varphi$ stands for $\boxed{1}\varphi$. A *literal* is either a propositional symbol or its negation; the set of literals is denoted by L. We denote by $\neg l$ the *complement* of the literal $l \in L$, that is, $\neg l$ denotes $\neg p$ if l is the propositional symbol p, and $\neg l$ denotes p if l is the literal $\neg p$. A *modal literal* is either $\boxed{a}l$ or $\neg\boxed{a}l$, where $l \in L$ and $a \in A_n$. The modal depth of a formula is recursively defined as follows:

Definition 2. *Let* $\varphi, \psi \in \mathsf{WFF}_{\mathsf{K}_n}$ *be well-formed formulae. We define* $\mathsf{mdepth} : \mathsf{WFF}_{\mathsf{K}_n} \longrightarrow \mathbb{N}$ *as* $\mathsf{mdepth}(p) = 0$*, for* $p \in P$*;* $\mathsf{mdepth}(\neg\varphi) = \mathsf{mdepth}(\varphi)$*;* $\mathsf{mdepth}(\varphi \wedge \psi) = \max(\mathsf{mdepth}(\varphi), \mathsf{mdepth}(\psi))$*; and* $\mathsf{mdepth}(\boxed{a}\varphi) = 1 + \mathsf{mdepth}(\varphi)$.

The modal level of a subformula is given relative to its position in the syntactic tree.

Definition 3. *Let* φ, φ' *be well-formed formulae. Let* Σ *be the alphabet* $\{1, 2, .\}$ *and* Σ^* *the set of all finite sequences over* Σ*. Denote by* ε *the empty sequence over* Σ*. Let* $\tau : \mathsf{WFF}_{\mathsf{K}_n} \times \Sigma^* \times \mathbb{N} \longrightarrow \mathscr{P}(\mathsf{WFF}_{\mathsf{K}_n} \times \Sigma^* \times \mathbb{N})$ *be the partial function inductively defined as follows (where* $\lambda \in \Sigma^*$*,* $ml \in \mathbb{N}$*):*

- $\tau(p, \lambda, ml) = \{(p, \lambda, ml)\}$*, for* $p \in P$*;*
- $\tau(\neg\varphi, \lambda, ml) = \{(\neg\varphi, \lambda, ml)\} \cup \tau(\varphi, \lambda.1, ml)$*;*
- $\tau(\boxed{a}\varphi, \lambda, ml) = \{(\boxed{a}\varphi, \lambda, ml)\} \cup \tau(\varphi, \lambda.1, ml + 1)$*;*
- $\tau(\varphi \wedge \varphi', \lambda, ml) = \{(\varphi \wedge \varphi', \lambda, ml)\} \cup \tau(\varphi, \lambda.1, ml) \cup \tau(\varphi', \lambda.2, ml)$.

The function τ applied to $(\varphi, \varepsilon, 0)$ returns the *annotated syntactic tree for* φ, where each node is uniquely identified by a subformula, its path order (or its position) in the tree, and its modal level. For instance, p occurs twice in the formula $\boxed{a}\boxed{a}(p \wedge \boxed{a}p)$, at the position 1.1.1 and modal level 2, and also at the position 1.1.2.1 and modal level 3.

Definition 4. *Let* φ *be a formula and let* $\tau(\varphi, \varepsilon, 0)$ *be its annotated syntactic tree. If* $(\varphi', \lambda', m') \in \tau(\varphi, \varepsilon, 0)$*, then* $\mathsf{mlevel}(\varphi, \varphi', \lambda) = m'$.

If $\mathsf{mlevel}(\varphi, \varphi', \lambda) = m$ we say that φ' at the position λ of φ occurs at the modal level m. In the example above, we have that p occurs at the modal levels 2 and 3.

We present the semantics of K_n, as usual, in terms of Kripke structures.

Definition 5. *A* Kripke model M for n agents over P *is given by a tuple* $(W, w_0, R_1, R_2, \ldots, R_n, \pi)$*, where* W *is a set of possible* worlds *with a distinguished world* w_0*, each* R_a *is a binary relation on* W*, and* $\pi : W \to (P \to \{true, false\})$ *is a function which associates to each world* $w \in W$ *a truth-assignment to propositional symbols.*

We write $\langle M, w\rangle \models \varphi$ (resp. $\langle M, w\rangle \not\models \varphi$) to say that φ is satisfied (resp. not satisfied) at the world w in the Kripke model M.

Definition 6. *Satisfaction of a formula at a given world* w *of a model* M *is inductively defined by:*

- $\langle M, w\rangle \models$ **true***;*
- $\langle M, w\rangle \models p$ *if, and only if,* $\pi(w)(p) = true$*, where* $p \in P$*;*
- $\langle M, w\rangle \models \neg\varphi$ *if, and only if,* $\langle M, w\rangle \not\models \varphi$*;*
- $\langle M, w\rangle \models (\varphi \wedge \psi)$ *if, and only if,* $\langle M, w\rangle \models \varphi$ *and* $\langle M, w\rangle \models \psi$*;*
- $\langle M, w\rangle \models \boxed{a}\varphi$ *if, and only if, for all* w'*,* wR_aw' *implies* $\langle M, w'\rangle \models \varphi$.

The formulae **false**, $(\varphi \lor \psi)$, $(\varphi \Rightarrow \psi)$, and $\diamondsuit_a \varphi$ are introduced as the usual abbreviations for $\neg$**true**, $\neg(\neg\varphi \land \neg\psi)$, $(\neg\varphi \lor \psi)$, and $\neg \boxed{a} \neg\varphi$, respectively. Let $M = (W, w_0, R_1, \ldots, R_n, \pi)$ be a model. For local satisfiability, formulae are interpreted with respect to the root of M, that is, w_0. A formula φ is *locally satisfied in* M, denoted by $M \models_L \varphi$, if $\langle M, w_0 \rangle \models \varphi$. The formula φ is *locally satisfiable* if there is a model M such that $\langle M, w_0 \rangle \models \varphi$. A formula φ is *globally satisfied in* M, if for all $w \in W$, $\langle M, w \rangle \models \varphi$. A formula φ is said to be *globally satisfiable* if there is a model M such that M globally satisfies φ, denoted by $M \models_G \varphi$. Satisfiability of sets of formulae is defined as usual.

When considering local satisfiability, the following holds (see, for instance, [9]):

Theorem 1. *Let $\varphi \in \mathsf{WFF}_{\mathsf{K}_n}$ be a formula and $M = (W, w_0, R_1, \ldots, R_n, \pi)$ be a model. $M \models_L \varphi$ if and only if there is a tree-like model M' such that $M' \models_L \varphi$. Moreover, M' is finite and its depth is bounded by* $\mathsf{mdepth}(\varphi)$.

Given a tree-like model $M = (W, w_0, R_1, \ldots, R_n, \pi)$, we denote by $\mathsf{depth}(w)$ the length of a path from w_0 to w through the union of the relations in M. The next result also holds.

Theorem 2. *Let $\varphi, \varphi' \in \mathsf{WFF}_{\mathsf{K}_n}$ and $M = (W, w_0, R_1, \ldots, R_n, \pi)$ be a tree-like model such that $M \models_L \varphi$. If $(\varphi', \lambda', m) \in \tau(\varphi, \varepsilon, 0)$ and φ' is satisfied in M, then there is $w \in W$, with $\mathsf{depth}(w) = m$, such that $\langle M, w \rangle \models \varphi'$. Moreover, the subtree rooted at w has height* $\mathsf{mdepth}(\varphi')$.

Theorem 2 is adapted from [2, Proposition 3.2]. The proof is by induction on the structure of a formula and shows that a subformula φ' of φ is satisfied at a node with distance m of the root of the tree-like model. As determining the satisfiability of a formula depends only on its subformulae, only the subtrees of height $\mathsf{mdepth}(\varphi')$ starting at level m need to be checked. The bound on the height of the subtrees follows from Theorem 1.

The global satisfiability problem for a (first-order definable) modal logic is equivalent to the local satisfiability problem of a logic obtained by adding the universal modality, $\boxed{*}$, to the original language [7]. Let K_n^* be the logic obtained by adding $\boxed{*}$ to K_n. Let $M = (W, w_0, R_1, \ldots, R_n, \pi)$ be a tree-like model for K_n. A model M^* for K_n^* is the pair (M, R_*), where $R_* = W \times W$. A formula $\boxed{*}\varphi$ is locally satisfied at the world w in the model M^*, written $\langle M^*, w \rangle \models_L \boxed{*}\varphi$, iff for all $w' \in W$, we have that $\langle M^*, w' \rangle \models \varphi$. Given these definitions, for φ in $\mathsf{WFF}_{\mathsf{K}_n}$, deciding $M \models_G \varphi$ is equivalent to deciding $M^* \models_L \boxed{*}\varphi$.

We note that although the full language of K_n^* enjoys the finite model property (it is satisfied in a model that is exponential in the size of the original formula [24]), it does not retain the finite tree model property. For instance, $\boxed{*}(p \Rightarrow \neg\boxed{*}p) \land \boxed{*}(\neg p \Rightarrow \neg\boxed{*}\neg p)$ cannot be satisfied in any finite tree-like structure [14]. The unravelling of a model M^* for K_n^* gives rise to an infinite tree-like model which satisfies the same formulae as M^*.

3 Layered Normal Form

A formula to be tested for local or global satisfiability is first translated into a normal form called *Separated Normal Form with Modal Levels*, SNF_{ml}. A formula in SNF_{ml}

is a conjunction of clauses labelled by the modal level in which they occur. We write $ml : \varphi$ to denote that φ occurs at the modal level $ml \in \mathbb{N} \cup \{*\}$. By $* : \varphi$ we mean that φ occurs at all modal levels. Formally, let $\mathsf{WFF}^{ml}_{\mathsf{K}_n}$ be the set of formulae $ml : \varphi$ such that $ml \in \mathbb{N} \cup \{*\}$ and $\varphi \in \mathsf{WFF}_{\mathsf{K}_n}$. Let $M^* = (W, w_0, R_1, \ldots, R_n, R_*, \pi)$ be a model and $\varphi \in \mathsf{WFF}_{\mathsf{K}_n}$. Satisfiability of labelled formulae is given by:

- $M^* \models_L ml : \varphi$ if, and only if, for all worlds $w \in W$ such that $\mathsf{depth}(w) = ml$, we have $\langle M^*, w \rangle \models_L \varphi$;
- $M^* \models_L * : \varphi$ if, and only if, $M^* \models_L \boxed{*}\, \varphi$.

Note that labels in a formula work as a kind of *weak* universal operator, allowing us to talk about formulae that are all satisfied at a given modal level.

Clauses in SNF_{ml} are in one of the following forms:

- Literal clause $\quad ml : \bigvee_{b=1}^{r} l_b$
- Positive a-clause $\quad ml : l' \Rightarrow \boxed{a}\, l$
- Negative a-clause $\quad ml : l' \Rightarrow \langle a \rangle\, l$

where $ml \in \mathbb{N} \cup \{*\}$ and $l, l', l_b \in L$. Positive and negative a-clauses are together known as *modal a-clauses*; the index a may be omitted if it is clear from the context.

Let φ be a formula in the language of K_n. In the following, we assume φ is in Negation Normal Form (NNF), that is, a formula where the operators are restricted to $\wedge$, $\vee$, $\boxed{a}$, $\langle a \rangle$ and $\neg$; also, only propositions are allowed in the scope of negations. The transformation of a formula φ into SNF_{ml} is achieved by recursively applying rewriting and renaming [18]. Let φ be a formula and t a propositional symbol not occurring in φ. For local satisfiability, the translation of φ is given by $0 : t \wedge \rho(0 : t \Rightarrow \varphi)$. We refer to clauses of the form $0 : D$, for a disjunction of literals D, as *initial clauses*. For global satisfiability, the translation of φ is given by $* : t \wedge \rho(* : t \Rightarrow \varphi)$ where t is a new propositional symbol. The translation function $\rho : \mathsf{WFF}^{ml}_{\mathsf{K}_n} \longrightarrow \mathsf{WFF}^{ml}_{\mathsf{K}_n}$ is defined as follows (with $\varphi, \varphi' \in \mathsf{WFF}_{\mathsf{K}_n}$, t' is a new propositional symbol, and $* + 1 = *$):

$$\begin{aligned}
\rho(ml : t \Rightarrow \varphi \wedge \varphi') &= \rho(ml : t \Rightarrow \varphi) \wedge \rho(ml : t \Rightarrow \varphi') \\
\rho(ml : t \Rightarrow \boxed{a}\varphi) &= (ml : t \Rightarrow \boxed{a}\varphi), \text{ if } \varphi \text{ is a literal} \\
&= (ml : t \Rightarrow \boxed{a}t') \wedge \rho(ml+1 : t' \Rightarrow \varphi), \text{ otherwise} \\
\rho(ml : t \Rightarrow \langle a \rangle\varphi) &= (ml : t \Rightarrow \langle a \rangle\varphi), \text{ if } \varphi \text{ is a literal} \\
&= (ml : t \Rightarrow \langle a \rangle t') \wedge \rho(ml+1 : t' \Rightarrow \varphi), \text{ otherwise} \\
\rho(ml : t \Rightarrow \varphi \vee \varphi') &= (ml : \neg t \vee \varphi \vee \varphi'), \text{ if } \varphi' \text{ is a disjunction of literals} \\
&= \rho(ml : t \Rightarrow \varphi \vee t') \wedge \rho(ml : t' \Rightarrow \varphi'), \text{ otherwise}
\end{aligned}$$

As the conjunction operator is commutative, associative, and idempotent, in the following we often refer to a formula in SNF_{ml} as a set of clauses. The next lemma shows that the transformation into SNF_{ml} is satisfiability preserving.

Lemma 1. *Let $\varphi \in \mathsf{WFF}_{\mathsf{K}_n}$ be a formula and let t be a propositional symbol not occurring in φ. Then: (1) φ is locally satisfiable if, and only if, $0 : t \wedge \rho(0 : t \Rightarrow \varphi)$ is satisfiable; (2) φ is globally satisfiable if, and only if, $* : t \wedge \rho(* : t \Rightarrow \varphi)$ is satisfiable.*

Proof. The *only if* part. For (1), if φ is locally satisfiable then there is a model M for K_n such that $M \models_L \varphi$. Let M^* be the model obtained from M by only adding the universal relation R_*. It is easy to check that M^* also locally satisfies φ. From Theorem 2, it follows that if a subformula of φ is satisfied, then it is satisfied at the modal level it occurs. In particular, we have that $M^* \models_L 0 : \varphi$. Thus, by induction on the structure of a formula together with the standard techniques related to renaming of formulae, we can build a model M'^* such that M'^* locally satisfies $0 : t \wedge \rho(0 : t \Rightarrow \varphi)$. For (2), if φ is globally satisfiable, then there is a model M for K_n such that $M \models_G \varphi$. Again, taking M^* as above, we have that $M^* \models_L \boxed{*}\,\varphi$ [7]. By the definition of satisfiability for labelled formulae, we have that $M^* \models_L * : \varphi$. By induction on the structure of a formula, by adding new literals as needed and properly setting their valuations at every world, we can build a model M'^* such that $M'^* \models_L * : t \wedge \rho(* : t \Rightarrow \varphi)$.

For the *if* part, let M^* be a model such that $M^* \models_L ml : t \wedge \rho(ml : t \Rightarrow \varphi)$, $ml = 0$ (resp. $ml = *$). The proof is standard: by ignoring the labels and the valuation of the propositional symbols not occurring in φ, it is easy to check that $M^* \models_L \varphi$ (resp. $M^* \models_L \boxed{*}\,\varphi$). From the results in [7], φ is locally (resp. globally) satisfiable. □

4 Inference Rules

The calculus comprises a set of inference rules for dealing with propositional and modal reasoning. In the following, we denote by σ the result of unifying the labels in the premises for each rule. Formally, unification is given by a function $\sigma : \mathscr{P}(\mathbb{N} \cup \{*\}) \longrightarrow \mathbb{N} \cup \{*\}$, where $\sigma(\{ml, *\}) = ml$; and $\sigma(\{ml\}) = ml$; otherwise, σ is undefined. The following inference rules can only be applied if the unification of their labels is defined (where $* - 1 = *$). Note that for GEN1 and GEN3, if the modal clauses occur at the modal level ml, then the literal clause occurs at the next modal level, $ml + 1$.

[LRES]

$$\begin{array}{r} ml : D \vee l \\ ml' : D' \vee \neg l \\ \hline \sigma(\{ml, ml'\}) : D \vee D' \end{array}$$

[MRES]

$$\begin{array}{r} ml : l_1 \Rightarrow \boxed{a}\, l \\ ml' : l_2 \Rightarrow \Diamond_{a} \neg l \\ \hline \sigma(\{ml, ml'\}) : \neg l_1 \vee \neg l_2 \end{array}$$

[GEN2]

$$\begin{array}{r} ml_1 : l'_1 \Rightarrow \boxed{a}\, l_1 \\ ml_2 : l'_2 \Rightarrow \boxed{a}\, \neg l_1 \\ ml_3 : l'_3 \Rightarrow \Diamond_{a} l_2 \\ \hline \sigma(\{ml_1, ml_2, ml_3\}) : \neg l'_1 \vee \neg l'_2 \vee \neg l'_3 \end{array}$$

[GEN1]

$$\begin{array}{l} ml_1 : l'_1 \Rightarrow \boxed{a}\, \neg l_1 \\ \quad\vdots \\ ml_m : l'_m \Rightarrow \boxed{a}\, \neg l_m \\ ml_{m+1} : l' \Rightarrow \Diamond_{a} \neg l \\ ml_{m+2} : l_1 \vee \ldots \vee l_m \vee l \\ \hline ml : \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l' \\ \text{where } ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\}) \end{array}$$

[GEN3]

$$\begin{array}{l} ml_1 : l'_1 \Rightarrow \boxed{a}\, \neg l_1 \\ \quad\vdots \\ ml_m : l'_m \Rightarrow \boxed{a}\, \neg l_m \\ ml_{m+1} : l' \Rightarrow \Diamond_{a} l \\ ml_{m+2} : l_1 \vee \ldots \vee l_m \\ \hline ml : \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l' \\ \text{where } ml = \sigma(\{ml_1, \ldots, ml_{m+1}, ml_{m+2} - 1\}) \end{array}$$

Definition 7. *Let Φ be a set of clauses in* SNF_{ml}*. A* derivation *from Φ is a sequence of sets $\Phi_0, \Phi_i, \ldots$ where $\Phi_0 = \Phi$ and, for each $i > 0$, $\Phi_{i+1} = \Phi_i \cup \{D\}$, where D is the resolvent obtained from Φ_i by an application of either* LRES*,* MRES*,* GEN1*,* GEN2*, or* GEN3*. We also require that D is in simplified form, $D \notin \Phi_i$, and that D is not a tautology. A* refutation *for Φ is a derivation $\Phi_0, \ldots, \Phi_k$, $k \in \mathbb{N}$, where ml :* **false** *$\in \Phi_k$, for $ml \in \mathbb{N} \cup \{*\}$.*

Before presenting correctness results, we show an example.

Example 1. Adapted from [1]. Clauses (1) and (2) say that a person is either female or male. Clauses (3) and (4), which are not genetically accurate, say that tall people have children with blond hair. The particular situation of Tom, denoted here by t_0, is given in the following clauses. Clauses (5), (6), and (7) say that Tom's daughters are tall. Clauses (8) and (9) say that Tom has a grandchild who is not blond. We want to prove that Tom has a son, which appears negated in Clause (10). The refutation is given below.

1. $* : female \vee male$	9. $1 : t_3 \Rightarrow \langle c \rangle \neg blond$	
2. $* : \neg female \vee \neg male$	10. $0 : t_0 \Rightarrow \boxed{c} \neg male$	
3. $* : \neg tall \vee t_1$	11. $1 : \neg t_1 \vee \neg t_3$	$[\mathrm{MRES}, 9, 4, blond]$
4. $* : t_1 \Rightarrow \boxed{c} blond$	12. $1 : \neg tall \vee \neg t_3$	$[\mathrm{LRES}, 11, 3, t_1]$
5. $0 : t_0$	13. $1 : \neg t_3 \vee \neg t_2 \vee \neg female$	$[\mathrm{LRES}, 7, 12, tall]$
6. $0 : t_0 \Rightarrow \boxed{c} t_2$	14. $1 : male \vee \neg t_2 \vee \neg t_3$	$[\mathrm{LRES}, 13, 1, tall]$
7. $1 : \neg t_2 \vee \neg female \vee tall$	15. $0 : \neg t_0$	$[\mathrm{GEN1}, 10, 6, 8, 14, male, t_2, t_3]$
8. $0 : t_0 \Rightarrow \langle c \rangle t_3$	16. $0 :$ **false**	$[\mathrm{LRES}, 15, 5, t_0]$

5 Correctness Results

In this section, we provide proofs for termination, soundness, and completeness of the calculus given in the previous section.

Theorem 3 (Termination). *Let Φ be a set of clauses in* SNF_{ml}*. Then, any derivation from Φ terminates.*

Proof. We regard a clause as a set of literals or modal literals. Let P_Φ be the set of propositional symbols occurring in Φ. We define $\overline{P_\Phi} = \{\neg p \mid p \in P_\Phi\}$, $L_\Phi = P_\Phi \cup \overline{P_\Phi}$, and $L_\Phi^{A_n} = \{\boxed{a} l, \langle a \rangle l \mid l \in L_\Phi \text{ and } a \in A_n\}$. As P_Φ and A_n are both finite and because none of the inference rules add new propositional symbols or new modal literals to the clause set, we have that $\mathscr{P}(L_\Phi \cup L_\Phi^{A_n})$ is finite and so it is the number of clauses that can be built from the symbols in P_Φ and A_n. □

Next, we show that the inference rules are sound.

Lemma 2 (LRES). *Let Φ be a set of clauses in* SNF_{ml} *with $\{ml : D \vee l, ml' : D' \vee \neg l\} \subseteq \Phi$. If Φ is satisfiable and $\sigma(\{ml, ml'\})$ is defined, then $\Phi \cup \{\sigma(\{ml, ml'\}) : D \vee D'\}$ is satisfiable.*

Proof. Let $M = (W, w_0, R_1, \ldots, R_n, R_*, \pi)$ be a model such that $M \models \Phi$. As $ml : D \vee l, ml' : D' \vee \neg l \in \Phi$, then $M \models ml : D \vee l$ and $M \models ml' : D' \vee \neg l$. Note that σ is commutative. Also, note that $\sigma(\{*, ml'\}) = ml'$. Finally, for $\sigma(\{ml, ml'\}) = *$, a particular modal level ml' is enough to show that the lemma holds. Hence, without loss of generality, assume $\sigma(\{ml, ml'\}) = ml'$. As $M \models ml' : D' \vee \neg l$, for all $w' \in W$ with $\mathsf{depth}(w') = ml'$, then $\langle M, w' \rangle \models D' \vee \neg l$. Similarly, because $\{ml, ml'\}$ is unifiable, from $M \models ml : D \vee l$, for all $w' \in W$ with $\mathsf{depth}(w') = ml'$, we obtain that $\langle M, w' \rangle \models D \vee l$. It follows that $\langle M, w' \rangle \models (D \vee l) \wedge (D' \vee \neg l)$, for all $w' \in W$ with $\mathsf{depth}(w') = ml'$. By soundness of resolution, $\langle M, w' \rangle \models D \vee D'$. As $\mathsf{depth}(w') = ml' = \sigma(\{ml, ml'\})$, we conclude that $M \models \sigma(\{ml, ml'\}) : D \vee D'$. □

Lemma 3 (MRES). *Let Φ be a set of clauses in SNF_{ml} with $\{ml : l_1 \Rightarrow \boxed{a} l, ml' : l_2 \Rightarrow \Diamond_a \neg l\} \subseteq \Phi$. If Φ is satisfiable and $\sigma(\{ml, ml'\})$ is defined, then $\Phi \cup \{\sigma(\{ml, ml'\}) : \neg l_1 \vee \neg l_2\}$ is satisfiable.*

Proof. The proof is similar to that of Lemma 2, as implications can be rewritten as disjunctions and $\Diamond_a \neg l$ is semantically equivalent to $\neg \boxed{a} l$. □

Lemma 4 (GEN1). *Let Φ be a set of clauses in SNF_{ml} with $\{ml_1 : l'_1 \Rightarrow \boxed{a} \neg l_1, \ldots, ml_m : l'_m \Rightarrow \boxed{a} \neg l_m, ml_{m+1} : l' \Rightarrow \Diamond_a \neg l, ml_{m+2} : l_1 \vee \ldots \vee l_m \vee l\} \subseteq \Phi$. If Φ is satisfiable and $\sigma(\{ml_1, \ldots, ml_m, ml_{m+1}, ml_{m+2} - 1\})$ is defined, then $\Phi \cup \{\sigma(\{ml_1, \ldots, ml_m, ml_{m+1}, ml_{m+2} - 1\}) : \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'\}$ is satisfiable.*

Proof. Let $M = (W, w_0, R_1, \ldots, R_n, R_*, \pi)$ be a model such that $M \models \Phi$. Note that if $* \in \{ml_1, \ldots, ml_m, ml_{m+1}, ml_{m+2} - 1\}$, then the formula labelled by $*$ holds at every world of the model and, therefore, at any modal level. Without loss of generality, assume $\sigma(\{ml_1, \ldots, ml_m, ml_{m+1}, ml_{m+2} - 1\}) = ml$, for a particular modal level ml. If Φ is satisfiable, then $M \models (ml : (l'_1 \Rightarrow \boxed{a} \neg l_1) \wedge \ldots \wedge (l'_m \Rightarrow \boxed{a} \neg l_m) \wedge (l' \Rightarrow \Diamond_a \neg l)) \wedge (ml + 1 : (l_1 \vee \ldots \vee l_m \vee l))$ and so for all worlds $w \in W$, with $\mathsf{depth}(w) = ml$, we have that **(1)** $\langle M, w \rangle \models (l'_1 \Rightarrow \boxed{a} \neg l_1) \wedge \ldots \wedge (l'_m \Rightarrow \boxed{a} \neg l_m) \wedge (l' \Rightarrow \Diamond_a \neg l)$. If $\langle M, w \rangle \not\models l'$, it follows easily that $\langle M, w \rangle \models \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'$ and, therefore, $M \models \sigma(\{ml_1, \ldots, ml_m, ml_{m+1}, ml_{m+2} - 1\}) : \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'$. The same occurs if any of the literals l'_i, $0 \leq i \leq m$, is not satisfied at w. We show, by contradiction, that this must be the case. Suppose $\langle M, w \rangle \models l'_1 \wedge \ldots \wedge l'_m \wedge l'$. From this and from (1), by the semantics of implication, the semantics of the modal operator $\Diamond_a$, and the semantics of the modal operator $\boxed{a}$, we have that there is a world w', with $\mathsf{depth}(w') = \mathsf{depth}(w) + 1$, where $\neg l_1 \wedge \ldots \wedge \neg l_m \wedge \neg l$ holds. Now, as $ml_{m+2} - 1$ is unifiable with $\{ml_1, \ldots, ml_m, ml_{m+1}\}$, for all worlds w'' with $\mathsf{depth}(w'') = \mathsf{depth}(w) + 1$, we obtain that $\langle M, w'' \rangle \models l_1 \vee \ldots \vee l_m \vee l$. In particular, because $\mathsf{depth}(w') = \mathsf{depth}(w'')$, we obtain that $\langle M, w' \rangle \models (\neg l_1 \wedge \ldots \wedge \neg l_m \wedge \neg l) \wedge (l_1 \vee \ldots \vee l_m \vee l)$. By several applications of the classical propositional resolution rule, $\langle M, w' \rangle \models$ **false**. This contradicts with the fact that Φ is satisfiable. Thus, $\langle M, w \rangle \models \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'$. As $\mathsf{depth}(w) = ml$, we conclude that $M \models \sigma(\{ml_1, \ldots, ml_m, ml_{m+1}, ml_{m+2} - 1\}) : \neg l'_1 \wedge \ldots \wedge \neg l'_m \wedge \neg l'$. □

Lemma 5 (GEN2). *Let Φ be a set of clauses in SNF_{ml} with $\{ml_1 : l'_1 \Rightarrow \boxed{a} l_1, ml_2 : l'_2 \Rightarrow \boxed{a} \neg l_1, ml_3 : l'_3 \Rightarrow \Diamond_a l_2\} \subseteq \Phi$. If Φ is satisfiable and $\sigma(\{ml_1, ml_2, ml_3\})$ is defined, then $\Phi \cup \{\sigma(\{ml_1, ml_2, ml_3\}) : \neg l'_1 \vee \neg l'_2 \vee \neg l'_3\}$ is satisfiable.*

Proof. From Lemma 4 by taking Φ such that $\{ml_1 : l'_1 \Rightarrow \boxed{a} l_1, ml_2 : l'_2 \Rightarrow \boxed{a} \neg l_1, ml_3 : l'_3 \Rightarrow \Diamond_a l_2, * : l_1 \vee \neg l_1 \vee \neg l_2\} \subseteq \Phi$, as $l_1 \vee \neg l_1 \vee \neg l_2$ is a tautology. □

Lemma 6 (GEN3). *Let Φ be a set of clauses in SNF_{ml} with $\{ml_1 : l'_1 \Rightarrow \boxed{a} \neg l_1, \ldots, ml_m : l'_m \Rightarrow \boxed{a} \neg l_m, ml_{m+1} : l' \Rightarrow \Diamond_a l, ml_{m+2} : l_1 \vee \ldots \vee l_m\} \subseteq \Phi$. If Φ is satisfiable and $\sigma(\{ml_1, \ldots, ml_m, ml_{m+1}, ml_{m+2} - 1\})$ is defined, then $\Phi \cup \{\sigma(\{ml_1, \ldots, ml_m, ml_{m+1}, ml_{m+2} - 1\}) : \neg l'_1 \vee \ldots \vee \neg l'_m \vee \neg l'\}$ is satisfiable.*

Proof. The formula $ml_{m+2} : l_1 \vee \ldots \vee l_m$ is semantically equivalent to $(ml_{m+2} : l_1 \vee \ldots \vee l_m \vee l) \wedge (ml_{m+2} : l_1 \vee \ldots \vee l_m \vee \neg l)$. The proof follows from Lemma 4 by taking Φ such that $\{ml_1 : l'_1 \Rightarrow \boxed{a} \neg l_1, \ldots, ml_m : l'_m \Rightarrow \boxed{a} \neg l_m, ml_{m+1} : l' \Rightarrow \Diamond_a l, ml_{m+2} : l_1 \vee \ldots \vee l_m \vee \neg l\} \subseteq \Phi$. □

Theorem 4 (Soundness). *Let Φ be a set of clauses in SNF_{ml} and $\Phi_0, \ldots, \Phi_k$, $k \in \mathbb{N}$, be a derivation for Φ. If Φ is satisfiable, then every Φ_i, $0 \leq i \leq k$, is satisfiable.*

Proof. From Lemmas 2-6, by induction on the number of sets in a derivation. □

Completeness is proved by showing that if a set T of clauses in SNF_{ml} is unsatisfiable, there is a refutation produced by the method presented here. The proof is by induction on the number of nodes of a graph, known as *behaviour graph*, built from T. Intuitively, nodes in the graph correspond to worlds and the set of edges correspond to the agents accessibility relations in a model. The graph construction is similar to the construction of a canonical model, followed by filtrations based on the set of clauses, often used to prove completeness for proof methods in modal logics [6]. Here, we first construct a graph G_G that satisfies the clauses labelled by $*$ and then complete the construction by unfolding G_G into a graph G which satisfies all clauses in T. We prove that an unsatisfiable set of clauses has an empty behaviour graph. In this case, there is a refutation using the inference rules given in Section 4.

Let T be a set of clauses in SNF_{ml}. Let $\{0, \ldots, m\}$ be the set of labels occurring in T. Formally, the behaviour graph G for n agents is a tuple $G = \langle N_0, \ldots, N_{m+1}, E_1, \ldots, E_n \rangle$, built from the set of SNF_{ml} clauses T, where N_i is a set of nodes for each modal level $0 \leq i \leq m+1$ occurring in T and each E_a is a set of edges labelled by $a \in A_n$. Every element of N_i is a set of literals and modal literals occurring in the modal level i in T. We require that nodes are *propositionally consistent sets*, i.e. they do not contain a (modal) literal and its negation. Note that $\Diamond_a \varphi$ is only an abbreviation for $\neg \boxed{a} \neg \varphi$. Thus, a set containing both $\boxed{a} \varphi$ and $\Diamond_a \neg \varphi$ is not propositionally consistent.

First, we define truth of a formula with respect to a set of literals and modal literals:

Definition 8. *Let V be a consistent set of literals and modal literals. Let φ, ψ, and ψ' be a Boolean combinations of literals and modal literals. We say that V satisfies φ (written $V \models \varphi$), if, and only if:*

- *$\varphi \in V$, if φ is a literal or a modal literal;*
- *φ is of the form $\psi \wedge \psi'$ and $V \models \psi$ and $V \models \psi'$;*
- *φ is of the form $\neg \psi$ and V does not satisfy ψ (written $V \not\models \psi$).*

A maximal consistent set of literals and modal literals contains either a propositional symbol or its negation; and it contains either a modal literal or its negation. We define satisfiability of a formula and a set of formulae with respect to a node:

Definition 9. *Let V be a maximal consistent set of literals and modal literals, η be a node in a behaviour graph G such that η consists of the literals and modal literals in V, φ be a Boolean combination of literals and modal literals, and $\chi = \{\varphi_1, \ldots, \varphi_m\}$ be a finite set of formulae, where each φ_i, $1 \leq i \leq m$, is a Boolean combination of literals and modal literals. We say that η* satisfies *φ (written $\eta \models \varphi$) if, and only if, $V \models \varphi$. We say that η* satisfies *χ (written $\eta \models \chi$) if, and only if, $\eta \models \varphi_1 \wedge \ldots \wedge \varphi_m$.*

The construction of the behaviour graph starts by partitioning a set of clauses T into two components corresponding to the set of global clauses and the set of local clauses. Let T_G be $\{* : \varphi \mid * : \varphi \in T\}$ and $T_L = T \setminus T_G$. First we construct a graph $G_G = \langle N, E'_1, \ldots, E'_n \rangle$, where N is the set of all maximal consistent sets of literals and modal literals occurring in T, that is, every node contains literals and modal literals that occur in T_G or T_L. Delete from N any nodes that do not satisfy D such that the literal clause $* : D$ is in T_G. This ensures that all literal clauses in T_G are satisfied at all nodes. If the set of nodes is empty, then the graph is empty and the literal clauses in T_G are unsatisfiable. Otherwise, the construction proceeds as follows. For all clauses $* : l' \Rightarrow \boxed{a} l$ (resp. $* : l' \Rightarrow \Diamond_a l$) in T_G, delete from N any nodes η such that $\eta \models l'$ and $\eta \not\models \boxed{a} l$ (resp. $\eta \not\models \Diamond_a l$). This ensures that all a-clauses in T_G are propositionally satisfied at N. We now construct the sets of edges related to each agent and ensure that any modal literal occurring in a node is also satisfied. Define E'_a as $N \times N$, which ensures that the tautology $\mathbf{true} \Rightarrow \boxed{a}\mathbf{true}$, for all $a \in A_n$, is satisfied at all nodes. For each clause $* : l' \Rightarrow \boxed{a} l$, delete from E'_a the edges (η, η') where $\eta \models \boxed{a} l$ but $\eta' \not\models l$. This ensures that all clauses of the form $* : l' \Rightarrow \boxed{a} l$ are now satisfied in G_G. Finally, repeatedly delete from G_G any nodes η such that $\eta \models \Diamond_a l$ and there is no η' such that $(\eta, \eta') \in E'_a$ and $\eta' \models l$. This ensures that all clauses of the form $* : l' \Rightarrow \Diamond_a l$ are now satisfied in G_G. If G_G is not empty, in order to satisfy the local constraints, given by clauses in T_L, we construct the graph G for T as follows.

Let $G_G = \langle N, E'_1, \ldots, E'_n \rangle$ be the non-empty graph for T_G constructed as above. Recall that $\{0, \ldots, m\}$ is the set of modal levels occurring as labels in T. The graph $G = \langle N_0, \ldots, N_{m+1}, E_1, \ldots, E_n \rangle$ for T is constructed by the unfolding of G_G as follows. Note that we need to construct the nodes at the level $m+1$ in order to satisfy the literals in the scope of modal operators at the level m. First, we construct the set of nodes N_{ml} for each modal level ml, $0 \leq ml \leq m+1$. Define $N_0 = N$, $N_{ml} = \emptyset$, for $0 < ml \leq m+1$, and $E_a = \emptyset$, for $0 \leq an$. For each $\eta, \eta' \in N$, if $\eta \in N_{ml}$ and $(\eta, \eta') \in E'_a$ for any $a \in A_n$, then add a copy of η', named η'_{ml+1}, to N_{ml+1} and make $E_a = E_a \cup \{(\eta, \eta'_{ml+1})\}$. For the highest modal level, $ml = m+1$, we also need to make sure that the global constraints are still satisfied: if $\eta \in N_{m+1}$ then we also add copies of all nodes reachable from η, by any relation E'_a, and add the corresponding relations to each E_a. Once the construction has finished, we delete nodes and edges in order to ensure that clauses in T are satisfied. Delete from N_{ml} any nodes that do not satisfy D for all literal clauses of the form $ml : D$ in T_L. Delete from N_{ml} any nodes that satisfy l', but do not satisfy $\boxed{a} l$ (resp. $\Diamond_a l$), for any modal clause, $ml : l' \Rightarrow \boxed{a} l$ (resp. $ml : l' \Rightarrow \Diamond_a l$). This ensures that all modal clauses are propositionally satisfied at every node in N_{ml}. Now, for every node η_{ml} in N_{ml}, delete a-edges as follows: if $ml : l' \Rightarrow \boxed{a} l \in T_L$ and $\eta_{ml} \models l'$, then delete any edges from η_{ml} to η'_{ml+1} labelled by a such that $\eta'_{ml+1} \not\models l$. This ensures that all positive a-clauses are satisfied by any nodes in G. Next, consider any nodes that do not satisfy the negative a-clauses in T_L and in T_G. For each node η_{ml} and for each agent $a \in A_n$, if $ml : l' \Rightarrow \Diamond_a l$ is in T_L or $* : l' \Rightarrow \Diamond_a l$ is in T_G, $\eta_{ml} \models l'$ and there is no a-edge between η_{ml} and a node at the level $ml+1$ that satisfies l, then η_{ml} is deleted. For the modal level $m+1$, we also consider the relations within this modal level when applying the deletion procedure. This ensures that all negative a-clauses are satisfied by all nodes $\eta_{ml} \in G$ at the modal level ml. If N_0 is empty, then the graph is empty.

The graph obtained after performing all possible deletions is called *reduced behaviour graph*. We show that a set of clauses is satisfiable if, and only if, the reduced graph for this set of clauses is non-empty.

Lemma 7. *Let T be a set of clauses. T is satisfiable if and only if the reduced behaviour graph G constructed from T is non-empty.*

Proof. ($\Rightarrow$) Assume that T is a satisfiable set of clauses. If we construct a graph from T, we generate a node for each maximal consistent set of literals and modal literals in T. Nodes are deleted only if they do not satisfy the set of literal clauses or the implications in modal clauses. The a-edges are constructed from each node to every other node, only deleting edges if the right-hand side of some positive a-clause is not satisfied. Similarly nodes are deleted if negative a-clauses cannot be satisfied. Hence a globally satisfiable set of clauses will result in a non-empty graph. If the graph is non-empty, the same procedure is applied at each modal level, ensuring that deletions are performed only if nodes and edges do not satisfy the set of clauses at that level. Thus a locally satisfiable set of clauses will also result in a non-empty graph.

($\Leftarrow$) Assume that the reduced graph $G = \langle N_0, \ldots, N_{m+1}, E_1, \ldots, E_n \rangle$ constructed from T is non-empty. To show that T is satisfiable we construct a model M from G. Let $ord : N_i \rightarrow \mathbb{N}$ be a total order on the nodes in N_i. Let $w_{ml,ord(\eta)}$ be the world named by $(ml, ord(\eta))$ for $\eta \in N_{ml}$ and let $W_{ml} = \bigcup_{\eta \in N_{ml}} \{w_{ml,ord(\eta)}\}$. The set of worlds W is given by $\bigcup_{i=0}^{m+1} W_i$. Let w_0 be any of the worlds in W_0. The pair $(w_{ml,ord(\eta)}, w_{ml',ord(\eta')})$ is in R_a if and only if $(\eta, \eta') \in E_a$. Also, take $R_* = W \times W$. Finally, set $\pi(w_{ml,ord(\eta)})(p) = true$ if and only if $p \in \eta$. This completes the construction of the model $M = (W, w_0, R_1, \ldots, R_n, R_*, \pi)$. □

We now show that the calculus for global and local reasoning in K_n is complete.

Theorem 5. *Let T be an unsatisfiable set of clauses in SNF_{ml}. Then there is a refutation for T by applying the resolution rules given in Section 4.*

Proof. Given a set of clauses T, construct the reduced behaviour graph as described above. First assume that the set of literal clauses is unsatisfiable. Thus all initial nodes will be removed from the reduced graph and the graph becomes empty. From the completeness of classical resolution there is a series of resolution steps which can be applied to these clauses which lead to the derivation of **false**. The same applies within any modal level. We can mimic these steps by applying the rule LRES to literal clauses and derive ml : **false**, for some modal level ml.

If the non-reduced graph is not empty and we have that both (1) $ml : l' \Rightarrow \boxed{a} l$ and (2) $ml' : l'' \Rightarrow \Diamond_a \neg l$ are in T, then, by construction of the graph, if $\{ml, ml'\}$ are unifiable, then any node in $N_{\sigma(\{ml,ml'\})}$ containing both l' and l'' is removed from the graph. The resolution rule MRES applied to (1) and (2) results in $\sigma(\{ml, ml'\}) : \neg l' \vee \neg l''$, simulating the deletion of nodes at the same modal level that satisfy both l' and l''.

Next, if the non-reduced graph is not empty, consider any nodes that do not satisfy the negative a-clauses in T. For each node $\eta_{ml} \in N_{ml}$ and for each agent $a \in A_n$, if $ml : l \Rightarrow \Diamond_{a} \neg l'$ is in T, $\eta_{ml} \models l$ and there is no a-edge between η and a node that satisfies $\neg l'$, then η_{ml} is deleted. We show next what inference rules or what inference steps correspond to the deletion of η_{ml}.

Let $\mathbb{C}_a^{\eta_{ml}}$ in T be the set of positive a-clauses corresponding to agent a, that is, the clauses of the form $ml : l_j \Rightarrow \boxed{a} l'_j$, where l_j and l'_j are literals, whose left-hand side are satisfied by η_{ml}. Let $\mathbb{R}_a^{\eta_{ml}}$ be the set of literals in the scope of $\boxed{a}$ on the right-hand side from the clauses in $\mathbb{C}_a^{\eta_{ml}}$, that is, if $ml : l_j \Rightarrow \boxed{a} l'_j \in \mathbb{C}_a^{\eta_{ml}}$, then $l'_j \in \mathbb{R}_a^{\eta_{ml}}$. From the construction of the graph, for a clause $ml : l \Rightarrow \Diamond_{a} l'$, if $\eta_{ml} \models l$ but there is no a-edge to a node containing l', it means that l', $\mathbb{R}_a^{\eta_{ml}}$, and the literal clauses at the level $ml+1$ must be contradictory. As l' alone is not contradictory and because the case where the literal clauses are contradictory by themselves has been covered above (by applications of LRES), there are five cases:

1. Assume that $\mathbb{R}_a^{\eta_{ml}}$ itself is contradictory. This means there must be clauses of the form $ml : l_1 \Rightarrow \boxed{a} l'', ml : l_2 \Rightarrow \boxed{a} \neg l'' \in \mathbb{C}_a^{\eta}$, where $\eta_{ml} \models l_1$ and $\eta_{ml} \models l_2$. Thus we can apply GEN2 to these clauses and the negative modal clause $ml : l \Rightarrow \Diamond_{a} l'$ deriving $ml : \neg l_1 \vee \neg l_2 \vee \neg l$. Hence the addition of this resolvent means that η_{ml} will be deleted as required.
2. Assume that l' and $\mathbb{R}_a^{\eta_{ml}}$ is contradictory. Then, $\mathbb{C}_a^{\eta_{ml}}$ in T contains a clause as $ml : l_1 \Rightarrow \boxed{a} \neg l'$ where, from the definition of $\mathbb{C}_a^{\eta_{ml}}$, $\eta_{ml} \models l_1$. Thus, by an application of MRES to this clause and $ml : l \Rightarrow \Diamond_{a} l'$, we derive $ml : \neg l_1 \vee \neg l$ and η_{ml} is removed as required.
3. Assume that l' and the literal clauses at the modal level $ml+1$ are contradictory. By consequence completeness of binary resolution [13], applications of LRES to the set of literal clauses generates $ml+1 : \neg l'$, which can be used together with $ml : l \Rightarrow \Diamond_{a} l'$ to apply GEN1 and generate $ml : \neg l$. This resolvent deletes η_{ml} as required. Note that this is a special case where the set of positive a-clauses in the premise of GEN1 is empty.
4. Assume that $\mathbb{R}_a^{\eta_{ml}}$ and the literal clauses at the modal level $m+1$ all contribute to the contradiction (but not l'), by the results in [13], applications of LRES will generate the relevant clause to which we can apply GEN3 and delete η_{ml} as required.
5. Assume that l', $\mathbb{R}_a^{\eta_{ml}}$ and the literal clauses all contribute to the contradiction. Thus, similarly to the above, applying LRES generates the relevant literal clause to which GEN1 can be applied. This deletes η_{ml} as required.

Summarising, LRES corresponds to deletions from the graph of nodes related to contradictions in the set of literal clauses at a particular modal level. The rule MRES also simulates classical resolution and corresponds to removing from the graph those nodes related to contradiction within the set of modal literals occurring at the same modal level. The inference rule GEN1 corresponds to deleting parts of the graph related to contradictions between the literal in the scope of $\Diamond_{a}$, the set of literal clauses, and the literals in the scope of $\boxed{a}$. The resolution rule GEN2 corresponds to deleting parts of the graph related to contradictions between the literals in the scope of $\boxed{a}$. Finally, GEN3 corresponds to deleting parts of the graph related to contradictions between the literals in the scope of $\boxed{a}$ and the set of literal clauses. These are all possible combinations of contradicting sets within a clause set.

If the resulting graph is empty, the set of clauses T is not satisfiable and there is a resolution proof corresponding to the deletion procedure, as described above. If the graph is not empty, by Lemma 7, a model for T can be built. □

6 Negative Resolution

Negative resolution was introduced in [19] as a refinement for the hyper-resolution method, which restricts the clauses that are candidates to being resolved. A literal is said to be negative if it is the negation of a propositional symbol. A clause is said to be negative if it contains only negative literals. Negative resolution can only be applied if one of the clauses being resolved is negative. Restricting the calculus given in Section 4 to negative resolution means that at least one of the literal clauses in the premises of inference rules is a negative clause. As it is, the calculus is not complete for negative resolution. However, it can be restricted to negative resolution with a small change in the normal form by allowing only positive literals in the scope of modal operators. Given a set of clauses in SNF_{ml}, we exhaustively apply the following rewriting rules (where $ml \in \mathbb{N} \cup \{*\}$, $t, p \in P$, and t' is a new propositional symbol):

$$\rho(ml : t \Rightarrow \boxed{a}\neg p) = (ml : t \Rightarrow \boxed{a}t') \wedge \rho(ml + 1 : t' \Rightarrow \neg p)$$
$$\rho(ml : t \Rightarrow \Diamond_a \neg p) = (ml : t \Rightarrow \Diamond_a t') \wedge \rho(ml + 1 : t' \Rightarrow \neg p)$$

It can be shown that the resulting set of clauses is satisfiable if, and only if, the original set of clauses is satisfiable. We call the resulting normal form SNF^{+}_{ml}. As the resulting set of clauses is still in SNF_{ml}, it follows immediately that the original calculus is terminating, sound, and complete for SNF^{+}_{ml}. Obviously, clause selection does not have any impact in soundness and termination. It rests to prove that restricting the application of the resolution rules to the case where at least one of the clauses is negative is complete.

Theorem 6. *Let Φ be a set of clauses in SNF^{+}_{ml}. If Φ is unsatisfiable, then there is a refutation from Φ by the negative version of the calculus given in Section 4.*

Proof. Under the new normal form, the inference rules MRES and GEN2 cannot be further applied. However, if there was a set of clauses in SNF_{ml} to which MRES (resp. GEN2) could be applied, then the set of clauses in SNF^{+}_{ml} contains sets of clauses to which GEN1 (resp. LRES and GEN3) can be applied. For sets of literal clauses, negative resolution is a complete strategy [19]. For the case where the literals in the scope of modal operators contradict with the set of literal clauses (cases 3, 4, and 5, in the proof of Theorem 5), the proof follows from the fact that negative resolution is also consequence complete [23]. Thus, the negative version of LRES still produces the negative clause needed for applying GEN1 and GEN3. □

Example 2. We show a negative refutation for the set of clauses given in Example 1. Clauses (9') and (10') are introduced in order to obtain a set of clauses in SNF^{+}_{ml}.

1. $* : female \vee male$	10. $0 : t_0 \Rightarrow \boxed{c} t_5$		
2. $* : \neg female \vee \neg male$	$10'$ $1 : \neg t_5 \vee \neg male$		
3. $* : \neg tall \vee t_1$	11. $1 : \neg t_3 \vee \neg t_1$	$[\mathsf{GEN1}, 4, 9, 9', b, t_4]$	
4. $* : t_1 \Rightarrow \boxed{c} blond$	12. $1 : \neg t_5 \vee female$	$[\mathsf{LRES}, 10', 1, male]$	
5. $0 : t_0$	13. $1 : \neg t_3 \vee \neg tall$	$[\mathsf{LRES}, 11, 3, t_1]$	
6. $0 : t_0 \Rightarrow \boxed{c} t_2$	14. $1 : \neg t_3 \vee \neg t_2 \vee \neg female$	$[\mathsf{LRES}, 7, 13, tall]$	
7. $1 : \neg t_2 \vee \neg female \vee tall$	15. $1 : \neg t_5 \vee \neg t_3 \vee \neg t_2$	$[\mathsf{LRES}, 14, 12, female]$	
8. $0 : t_0 \Rightarrow \diamondsuit_c t_3$	16. $0 : \neg t_0$	$[\mathsf{GEN1}, 10, 6, 8, 15, t_5, t_2, t_3]$	
9. $1 : t_3 \Rightarrow \diamondsuit_c t_4$	17. $0 : \mathbf{false}$	$[\mathsf{LRES}, 5, 16, t_0]$	
$9'$. $2 : \neg t_4 \vee \neg blond$			

7 Ordered Resolution

Ordered resolution is a refinement of resolution where inferences are restricted to maximal literals in a clause, with respect to a well-founded ordering on literals. Formally, let Φ be a set of clauses and P_Φ be the set of propositional symbols occurring in Φ. Let $\succ$ be a well-founded and total ordering on P_Φ. This ordering can be extended to literals L_Φ occurring in Φ by setting $\neg p \succ p$ and $p \succ \neg q$ whenever $p \succ q$, for all $p, q \in P_\Phi$. A literal l is said to be *maximal* with respect to a clause $C \vee l$ if, and only if, there is no l' occurring in C such that $l' \succ l$. In the case of classical binary resolution, the ordering refinement restricts the application to clauses $C \vee l$ and $D \vee l'$ where l is maximal with respect to C and l' is maximal with respect to D. Ordered resolution is refutational complete [10] and it has been successfully applied as the core strategy for many automated tools for both classical and modal logics [21,25,26,4,12]. It has also been shown that classical hyper-resolution is complete under ordering refinements for any ordering on the set of literals [5]. Restricting resolution by admissible orderings has been proved complete for hybrid logics as well [3].

We show that the restriction given by ordered resolution cannot be easily applied to the calculus given in Section 4. Orderings can be used to find contradictions at the propositional fragment of the language by restricting the application of LRES. However, the application of the hyper-resolution rules (GEN1, GEN2, and GEN3) require a consequence complete procedure, so that the relevant literal clauses for applying those inference rules are generated. As ordered resolution lacks consequence completeness [15], the resulting restricted calculus is not complete either. Consider the following set of clauses:

1. $0 : t_0$	3. $1 : a \vee \neg t_1$	5. $0 : t_0 \Rightarrow \Diamond t_2$
2. $0 : t_0 \Rightarrow \Box t_1$	4. $1 : b \vee \neg t_1$	6. $1 : \neg a \vee \neg b \vee \neg t_2$

which is clearly unsatisfiable. The ordering given by $t_0 \succ t_1 \succ t_2 \succ a \succ b$ does not allow any inference rule to be applied. Reversing the ordering allows a refutation to be found for this particular example. One might conjecture that imposing an ordering where the original literals in a clause are maximal with respect to the literals introduced by renaming might result in a complete calculus. That is not the case. Consider the next example where such an ordering has been used. Literals are ordered within each clause, that is, the rightmost literal is the maximal literal with respect to each clause.

1. $0 : t_0 \Rightarrow \Diamond t_1$	5. $1 : \neg t_3 \vee \neg t_1$
2. $0 : t_0 \Rightarrow \Box t_2$	6. $1 : \neg t_2 \vee \neg t_1 \vee \neg p$ $[LRES, 4, 5, t_3]$
3. $1 : \neg t_1 \vee p$	7. $1 : \neg t_2 \vee \neg t_1$ $[LRES, 6, 3, p]$
4. $1 : t_3 \vee \neg t_2 \vee \neg p$	

As shown, the negative version of the calculus is able to find a refutation by resolving Clauses (4) and (5) on the literal t_3, producing Clause (6), and by further applying LRES, obtaining Clause (7) to which GEN1 can be applied. Ordered resolution will not produce Clause (6) nor a clause that subsumes it. Thus, a refutation is not found.

8 Conclusion

We have presented a complete calculus based on modal levels for both local and global reasoning for the multimodal basic propositional modal logic, K_n. We have also shown that, by a small change in the normal form, negative resolution is complete, reducing the search space and also reducing the number of inference rules. Finally, we established that ordered resolution is not complete for the given calculus. Determining an admissible ordering that would make the restricted calculus complete does not seem to be a trivial task. We conjecture that the use of an appropriate selection function, as given in [17] in the context of disjunctive modal programs, might lead to a complete calculus. Other strategies, as those considered in [22], which combines hyper-tableaux and ordered resolution to deal with reasoning tasks for Description Logics, are also subject of future work. The implementation of the calculus given here as well as the experimental evaluation and performance comparison with other reasoners are current work.

Acknowledgments The authors would like to thank the anonymous reviewers for their suggestions and further references.

References

1. Areces, C., de Nivelle, H., de Rijke, M.: Prefixed Resolution: A Resolution Method for Modal and Description Logics. In: Ganzinger, H. (ed.) Proc. CADE-16. LNAI, vol. 1632, pp. 187–201. Springer, Berlin (Jul 7–10 1999)
2. Areces, C., Gennari, R., Heguiabehere, J., Rijke, M.D.: Tree-based heuristics in modal theorem proving. In: In Proc. of ECAI 2000. p. pages. IOS Press (2000)

3. Areces, C., Gorn, D.: Resolution with order and selection for hybrid logics. Journal of Automated Reasoning 46(1), 1–42 (2011)
4. Areces, C., Heguiabehere, J.: HyLoRes: A hybrid logic prover (Sep 18 2002), http://citeseer.ist.psu.edu/619572.html;http://turing.wins.uva.nl/~carlos/Papers/aiml2002-2.ps.gz
5. Bachmair, L., Ganzinger, H.: On restrictions of ordered paramodulation with simplification. In: Proc. CADE-10. pp. 427–441. LNCS, Springer-Verlag, Berlin (1990)
6. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic. Cambridge University Press, Cambridge, England (2001)
7. Goranko, V., Passy, S.: Using the universal modality: gains and questions. Journal of Logic and Computation 2(1), 5–30 (1992)
8. Hähnle, R.: Automated Deduction in Multiple-Valued Logics. Oxford University Press (1993)
9. Halpern, J.Y., Moses, Y.: A guide to completeness and complexity for modal logics of knowledge and belief. Artificial Intelligence 54(3), 319–379 (Apr 1992)
10. Hayes, P.J., Kowalski, R.A.: Semantic trees in automatic theorem proving. In: Meltzer, B., Michie, D. (eds.) Machine Intelligence, 5, pp. 87–101. Elsevier, New York, New York (1969)
11. Horrocks, I.R., Hustadt, U., Sattler, U., Schmidt, R.: Computational modal logic. In: Blackburn, P., van Benthem, J., Wolter, F. (eds.) Handbook of Modal Logic, pp. 181–245. Elsevier Science Inc., New York (2006)
12. Hustadt, U., Konev, B.: TRP++: A temporal resolution prover. In: Baaz, M., Makowsky, J., Voronkov, A. (eds.) Collegium Logicum, pp. 65–79. Kurt Gödel Society (2004)
13. Lee, R.C.T.: A completeness theorem and computer proram for finding theorems derivable from given axioms. Ph.D. thesis, Berkeley (1967)
14. Marx, M., Venema, Y.: Local variations on a loose theme: Modal logic and decidability. In: Finite Model Theory and Its Applications, pp. 371–429. Texts in Theoretical Computer Science an EATCS Series, Springer Berlin Heidelberg (2007)
15. Minicozzi, E., Reiter, R.: A note on linear resolution strategies in consequence-finding. Artificial Intelligence 3(1–3), 175–180 (1972)
16. Nalon, C., Dixon, C.: Clausal resolution for normal modal logics. J. Algorithms 62, 117–134 (July 2007), http://dl.acm.org/citation.cfm?id=1316091.1316347
17. Nguyen, L.A.: Negative ordered hyper-resolution as a proof procedure for disjunctive logic programming. Fundam. Inform 70(4), 351–366 (2006)
18. Plaisted, D.A., Greenbaum, S.A.: A Structure-Preserving Clause Form Translation. Journal of Logic and Computation 2, 293–304 (1986)
19. Robinson, J.A.: Automatic Deduction with Hyper-resolution. International Journal of Computer Mathematics 1, 227–234 (1965)
20. Schild, K.: A Correspondence Theory for Terminological Logics. In: Proceedings of the 12th IJCAI. pp. 466–471. Springer (1991)
21. Schulz, S.: The E theorem prover (2013), http://wwwlehre.dhbw-stuttgart.de/~sschulz/E/E.html
22. Simancik, F., Motik, B., Horrocks, I.: Consequence-based and fixed-parameter tractable reasoning in description logics. Artif. Intell 209, 29–77 (2014)
23. Slagle, J.R., Chang, C.L., Lee, R.C.T.: Completeness theorems for semantic resolution in consequence-finding. In: Proceedings of the 1st IJCAI. pp. 281–286. William Kaufmann (May 1969)
24. Spaan, E.: Complexity of Modal Logics. Ph.D. thesis, University of Amsterdam (1993)
25. The SPASS Team: Automation of logic: Spass (2010), http://www.spass-prover.org/
26. Voronkov, A.: Vampire, http://www.vprover.org/index.cgi