Clausal Resolution for Modal Logics of Confluence

Cláudia Nalon[1,*], João Marcos[2,**], and Clare Dixon[3]

[1] Departament of Computer Science, University of Brasília
C.P. 4466 – CEP:70.910-090 – Brasília – DF – Brazil
`nalon@unb.br`
[2] LoLITA and Dept. of Informatics and Applied Mathematics, UFRN, Brazil
`jmarcos@dimap.ufrn.br`
[3] Department of Computer Science, University of Liverpool
Liverpool, L69 3BX – United Kingdom
`CLDixon@liverpool.ac.uk`

Abstract. We present a clausal resolution-based method for normal multimodal logics of confluence, whose Kripke semantics are based on frames characterised by appropriate instances of the Church-Rosser property. Here we restrict attention to eight families of such logics. We show how the inference rules related to the normal logics of confluence can be systematically obtained from the parametrised axioms that characterise such systems. We discuss soundness, completeness, and termination of the method. In particular, completeness can be modularly proved by showing that the conclusions of each newly added inference rule ensures that the corresponding conditions on frames hold. Some examples are given in order to illustrate the use of the method.

Keywords: normal modal logics, combined logics, resolution method.

1 Introduction

Modal logics are often introduced as extensions of classical logic with two additional unary operators: "$\Box$" and "$\Diamond$", whose meanings vary with the field of application to which they are tailored to apply. In the most common interpretation, formulae "$\Box p$" and "$\Diamond p$" are read as "p is necessary" and "p is possible", respectively. Evaluation of a modal formula depends upon an organised collection of scenarios known as *possible worlds*. Different modal logics assume different *accessibility relations* between such worlds. Worlds and their accessibility relations define a so-called *Kripke frame*. The evaluation of a formula hinges on such structure: given an appropriate accessibility relation and a world w, a formula $\Box p$ is satisfied at w if p is true at all worlds accessible from w; a formula $\Diamond p$ is satisfied at w if p is true at some world accessible from w.

In normal modal logics extending the classical propositional logic, the schema $\Box(\varphi \Rightarrow \psi) \Rightarrow (\Box\varphi \Rightarrow \Box\psi)$ (the distribution axiom **K**), where φ and ψ are well-formed formulae and $\Rightarrow$ stands for classical implication, is valid, and the schematic

rule $\varphi/\Box\varphi$ (the necessitation rule **Nec**) preserves validity. The weakest of these logics, named $\mathsf{K}_{(1)}$, is semantically characterised by the class of Kripke frames with no restrictions imposed on the accessibility relation. In the multimodal version, named $\mathsf{K}_{(n)}$, Kripke frames are directed multigraphs and modal operators are equipped with indexes over a set of *agents*, given by $\mathcal{A}_n = \{1, 2, \ldots, n\}$, for some positive integer n. Accordingly, in this case classical logic is extended with operators $\boxed{1}, \boxed{2}, \ldots, \boxed{n}$, where a formula as $\boxed{a}p$, with $a \in \mathcal{A}_n$, may be read as "agent a considers p to be necessary". The modal operator $\Diamond\!\!\!\Diamond$ is the dual of $\boxed{a}$, being introduced as an abbreviation for $\neg\boxed{a}\neg$, where $\neg$ stands for classical negation. The logic $\mathsf{K}_{(n)}$ can be seen as the *fusion* of n copies of $\mathsf{K}_{(1)}$ and its axiomatisation is given by the union of the axioms for classical propositional logic with the axiomatic schemata $\mathbf{K}_a$, namely $\boxed{a}(\varphi \Rightarrow \psi) \Rightarrow (\boxed{a}\varphi \Rightarrow \boxed{a}\psi)$, for each $a \in \mathcal{A}_n$; and the set of inference rules is given by *modus ponens* and the rule schemata $\mathbf{Nec}_a$, namely $\varphi/\boxed{a}\varphi$, for each $a \in \mathcal{A}_n$.

The basic normal multimodal logic $\mathsf{K}_{(n)}$ and its extensions have been widely used to represent and reason about complex systems. Some of the interesting extensions include the normal multimodal logics based on $\mathbf{K}_a$ and (the combination of) axioms as, for instance, $\mathbf{T}_a$ ($\boxed{a}\varphi \Rightarrow \varphi$), $\mathbf{D}_a$ ($\boxed{a}\varphi \Rightarrow \Diamond\!\!\!\Diamond\varphi$), $\mathbf{4}_a$ ($\boxed{a}\varphi \Rightarrow \boxed{a}\boxed{a}\varphi$), $\mathbf{5}_a$ ($\Diamond\!\!\!\Diamond\varphi \Rightarrow \boxed{a}\Diamond\!\!\!\Diamond\varphi$), and $\mathbf{B}_a$ ($\Diamond\!\!\!\Diamond\boxed{a}\varphi \Rightarrow \varphi$). For example, the description logic $\mathcal{ALC}$, which is employed for reasoning about ontologies, is a syntactic variant of $\mathsf{K}_{(1)}$ [22]; the epistemic logic, denoted by $\mathsf{S5}_{(n)}$, which is used in dealing with problems ranging from multi-agency to communication protocols [21,11], can be axiomatised by combining $\mathbf{K}_a$, $\mathbf{T}_a$, and $\mathbf{5}_a$. The addition of those axioms (or their combinations) to $\mathsf{K}_{(n)}$ imposes some restrictions on the class of models where formulae are valid. Thus, a formula valid in a logic containing $\mathbf{T}_a$ is valid only if it is valid in a frame where the accessibility relation for each agent a is *reflexive*. The other axioms, $\mathbf{D}_a$, $\mathbf{4}_a$, $\mathbf{5}_a$, and $\mathbf{B}_a$, demand the accessibility relation for each agent a to be, respectively, *serial*, *transitive*, *Euclidean*, and *symmetric*.

A *logic of confluence* $\mathsf{K}_{(n)}^{p,q,r,s}$ is a modal system axiomatised by $\mathsf{K}_{(n)}$ plus axioms $\mathbf{G}_a^{p,q,r,s}$ of the form

$$\Diamond\!\!\!\Diamond_a^p \boxed{a}^q \varphi \Rightarrow \boxed{a}^r \Diamond\!\!\!\Diamond^s \varphi$$

where $a \in \mathcal{A}_n$, φ is a well-formed formula, $p, q, r, s \in \mathbb{N}$, where $\boxed{a}^0\varphi \stackrel{\text{def}}{=} \varphi$ and $\boxed{a}^{i+1}\varphi \stackrel{\text{def}}{=} \boxed{a}\boxed{a}^i\varphi$, and where $\Diamond\!\!\!\Diamond^0\varphi \stackrel{\text{def}}{=} \varphi$ and $\Diamond\!\!\!\Diamond^{i+1}\varphi \stackrel{\text{def}}{=} \Diamond\!\!\!\Diamond\Diamond\!\!\!\Diamond^i\varphi$, for $i \in \mathbb{N}$ (the superscript is often omitted if equal to 1). Such axiomatic schemata were notably studied by Lemmon [16]. Using Modal Correspondence Theory, it can be shown that the frame condition on a logic where an instance of $\mathbf{G}_a^{p,q,r,s}$ is valid corresponds to a generalised diamond-like structure representing the *Church-Rosser property* (the philosophical literature sometimes calls such property 'incestual' [8]), as illustrated in Fig. 1 [6]. To be more precise, let $\mathcal{W}$ be a nonempty set of worlds and let $\mathcal{R}_a \subseteq \mathcal{W} \times \mathcal{W}$ be the accessibility relation of agent $a \in \mathcal{A}_n$. By $w\mathcal{R}_a^0 w'$ we mean that $w = w'$, and $w\mathcal{R}_a^{i+1}w'$ means that there is some world w'' such that $w\mathcal{R}_a w''$ and $w''\mathcal{R}_a^i w'$. Thus, $w\mathcal{R}_a^i w'$ holds if there is an i-long $\mathcal{R}_a$-path from w to w'; alternatively, to assert that, we may also write $(w, w') \in \mathcal{R}_a^i$. Given these definitions, the condition on frames that corresponds to the axiom $\mathbf{G}_a^{p,q,r,s}$ is described by $\forall w_0, w_1, w_2$ $(w_0\mathcal{R}_a^p w_1 \land w_0\mathcal{R}_a^r w_2 \Rightarrow \exists w_3(w_1\mathcal{R}_a^q w_3 \land w_2\mathcal{R}_a^s w_3))$, where $w_0, w_1, w_2, w_3 \in \mathcal{W}$.
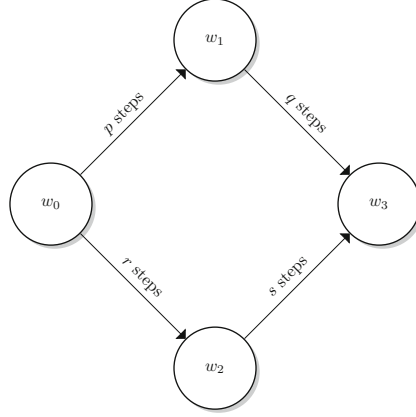
Fig. 1. Church-Rosser property for frames where $\mathbf{G}_a^{p,q,r,s} = \Diamond^p \Box^q \varphi \Rightarrow \Box^r \Diamond^s \varphi$ is valid

Many well-known modal axiomatic systems are identified with particular logics of confluence. For instance, $\mathsf{T}_{(n)}$ corresponds to $\mathsf{K}_{(n)}^{0,1,0,0}$, a normal modal logic in which the axiom $\Box \varphi \Rightarrow \varphi$ is valid, for all $a \in \mathcal{A}_n$ and any formula φ. The axiom $\mathbf{4}_a$ may be written as $\mathbf{G}_a^{0,1,2,0}$, that is, $\Box^1 \varphi \Rightarrow \Box^2 \varphi$. The Geach axiom $\mathbf{G1}_a$ is given by $\mathbf{G}_a^{1,1,1,1}$ ($\Diamond \Box \varphi \Rightarrow \Box \Diamond \varphi$). Formulae in $\mathsf{K}_{(n)}^{1,1,1,1}$ are satisfiable if, and only if, they are satisfiable in a model with n relations satisfying the so-called 'diamond property', and analogous claims hold for instance concerning formulae of $\mathsf{T}_{(n)}$ and models whose relations are all reflexive, and formulae of $\mathsf{4}_{(n)}$ and models whose relations are all transitive.

Logics of confluence are interesting not only because they encompass a great number of normal modal logics as particular examples, but also in view of their attractive computational behaviour. Indeed, if we think of multimodal frames as abstract rewriting systems, for instance, and think of modal languages as a way of obtaining an internal and local perspective on such frames, then each given notion of confluence ensures that certain different paths of transformation will eventually lead to the same result. Having a decidable proof procedure for a logic underlying such class of frames helps in establishing a direct form of verifying the properties of the structures that they represent.

As a contribution towards a uniform approach to the development of proof methods for logics of confluence, in this work we deal with the logics where $p, q, r, s \in \{0, 1\}$. Table 1 shows the relevant axiomatic schemata, some standard names by which they are known, and the corresponding conditions on frames. The axiom $\mathbf{G}_a^{0,1,1,1}$ seems not to be named in the literature; the corresponding property follows the naming convention given in [5, pg. 127]. Note that $\mathbf{G}_a^{0,0,0,0}$, $\mathbf{G}_a^{0,1,1,0}$, and $\mathbf{G}_a^{1,0,0,1}$ are instances of classical tautologies and are thus not included in Table 1. Also, given the duality between $\Box$ and $\Diamond$, $\mathbf{G}_a^{p,q,r,s}$ is semantically equivalent to $\mathbf{G}_a^{r,s,p,q}$. Thus, there are in fact eight families of multimodal logics related to the axioms $\mathbf{G}_a^{p,q,r,s}$, where $p, q, r, s \in \{0, 1\}$.

We present a clausal resolution-based method for solving the satisfiability problem in logics axiomatised by $\mathbf{K}_a$ plus $\mathbf{G}_a^{p,q,r,s}$, where $p, q, r, s \in \{0, 1\}$. The resolution

Table 1. Axioms and corresponding conditions on frames

(p,q,r,s)	Name	Axioms	Property	Condition on Frames
$(0,0,1,1)$ $(1,1,0,0)$	$\mathbf{B}_a$	$\varphi \Rightarrow \boxed{a} \Diamond\varphi$ $\Diamond \boxed{a}\varphi \Rightarrow \varphi$	symmetric	$\forall w, w'(w\mathcal{R}_a w' \Rightarrow w'\mathcal{R}_a w)$
$(0,0,1,0)$ $(1,0,0,0)$	$\mathbf{Ban}_a$	$\varphi \Rightarrow \boxed{a}\varphi$ $\Diamond\varphi \Rightarrow \varphi$	modally banal	$\forall w, w'(w\mathcal{R}_a w' \Rightarrow w = w')$
$(0,1,0,1)$	$\mathbf{D}_a$	$\boxed{a}\varphi \Rightarrow \Diamond\varphi$	serial	$\forall w \exists w'(w\mathcal{R}_a w')$
$(1,0,1,0)$	$\mathbf{F}_a$	$\Diamond\varphi \Rightarrow \boxed{a}\varphi$	functional	$\forall w, w', w''((w\mathcal{R}_a w' \wedge w\mathcal{R}_a w'') \Rightarrow w' = w'')$
$(0,0,0,1)$ $(0,1,0,0)$	$\mathbf{T}_a$	$\varphi \Rightarrow \Diamond\varphi$ $\boxed{a}\varphi \Rightarrow \varphi$	reflexive	$\forall w(w\mathcal{R}_a w)$
$(1,0,1,1)$ $(1,1,1,0)$	$\mathbf{5}_a$	$\Diamond\varphi \Rightarrow \boxed{a} \Diamond\varphi$ $\Diamond \boxed{a}\varphi \Rightarrow \boxed{a}\varphi$	Euclidean	$\forall w, w', w''((w\mathcal{R}_a w' \wedge w\mathcal{R}_a w'') \Rightarrow w'\mathcal{R}_a w'')$
$(1,1,1,1)$	$\mathbf{G1}_a$	$\Diamond \boxed{a}\varphi \Rightarrow \boxed{a} \Diamond\varphi$	convergent	$\forall w, w', w''((w\mathcal{R}_a w' \wedge w\mathcal{R}_a w'') \Rightarrow \exists w'''(w'\mathcal{R}_a w''' \wedge w''\mathcal{R}_a w'''))$
$(0,1,1,1)$ $(1,1,0,1)$	$\mathbf{G}_a^{0,1,1,1}$	$\boxed{a}\varphi \Rightarrow \boxed{a} \Diamond\varphi$ $\Diamond \boxed{a}\varphi \Rightarrow \Diamond\varphi$	0,1,1,1-convergent	$\forall w, w'(w\mathcal{R}_a w' \Rightarrow \exists w''(w\mathcal{R}_a w'' \wedge w'\mathcal{R}_a w''))$

calculus is based on that of [17], which deals with the logical fragment corresponding to $\mathsf{K}_{(n)}$. The new inference rules to deal with axioms of the form $\mathbf{G}_a^{p,q,r,s}$ add relevant information to the set of clauses: the conclusion of each inference rule ensures that properties related to the corresponding conditions on frames hold, that is, the newly added clauses capture the required properties of a model. We discuss soundness, completeness, and termination. Full proofs can be found in [18].

2 The Normal Modal Logic $\mathsf{K}_{(n)}$

The set $\mathsf{WFF}_{\mathsf{K}_{(n)}}$ of *well-formed formulae* of the logic $\mathsf{K}_{(n)}$ is constructed from a denumerable set of *propositional symbols*, $\mathcal{P} = \{p, q, p', q', p_1, q_1, \ldots\}$, the negation symbol $\neg$, the conjunction symbol $\wedge$, the propositional constant **true**, and a unary connective $\boxed{a}$ for each agent a in the finite set of agents $\mathcal{A}_n = \{1, \ldots, n\}$. When $n = 1$, we often omit the index, that is, $\Box\varphi$ stands for $\boxed{1}\varphi$. As usual, $\Diamond$ is introduced as an abbreviation for $\neg\boxed{a}\neg$. A *literal* is either a propositional symbol or its negation; the set of literals is denoted by $\mathcal{L}$. By $\neg l$ we will denote the *complement* of the literal $l \in \mathcal{L}$, 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 $l \in \mathcal{L}$ and $a \in \mathcal{A}_n$.

We present the semantics of $\mathsf{K}_{(n)}$, as usual, in terms of Kripke frames.

Definition 1. *A* Kripke frame $\mathcal{S}$ *for* n *agents over* $\mathcal{P}$ *is given by a tuple* $(\mathcal{W}, w_0, \mathcal{R}_1, \mathcal{R}_2, \ldots, \mathcal{R}_n)$, *where* $\mathcal{W}$ *is a set of possible* worlds *(or* states*) with a distinguished world* w_0, *and each* $\mathcal{R}_a$ *is a binary relation on* $\mathcal{W}$. *A* Kripke model $\mathcal{M} = (\mathcal{S}, \pi)$ *equips a Kripke frame* $\mathcal{S}$ *with a function* $\pi : \mathcal{W} \to (\mathcal{P} \to \{$true$,$false$\})$

that plays the role of an interpretation that associates to each state $w \in \mathcal{W}$ a truth-assignment to propositional symbols.

The so-called accessibility relation $\mathcal{R}_a$ is a binary relation that captures the notion of relative possibility from the viewpoint of agent a: A pair (w, w') is in $\mathcal{R}_a$ if agent a considers world w' possible, given the information available to her in world w. We write $\langle \mathcal{M}, w \rangle \models \varphi$ (resp. $\langle \mathcal{M}, w \rangle \not\models \varphi$) to say that φ is satisfied (resp. not satisfied) at the world w in the Kripke model $\mathcal{M}$.

Definition 2. *Satisfaction of a formula at a given world w of a model $\mathcal{M}$ is set by:*

- $\langle \mathcal{M}, w \rangle \models$ **true**
- $\langle \mathcal{M}, w \rangle \models p$ *if, and only if, $\pi(w)(p) = true$, where $p \in \mathcal{P}$*
- $\langle \mathcal{M}, w \rangle \models \neg\varphi$ *if, and only if, $\langle \mathcal{M}, w \rangle \not\models \varphi$*
- $\langle \mathcal{M}, w \rangle \models (\varphi \wedge \psi)$ *if, and only if, $\langle \mathcal{M}, w \rangle \models \varphi$ and $\langle \mathcal{M}, w \rangle \models \psi$*
- $\langle \mathcal{M}, w \rangle \models \boxed{a}\varphi$ *if, and only if $\langle \mathcal{M}, w' \rangle \models \varphi$, for all w' such that $w\mathcal{R}_a w'$*

The formulae **false**, $(\varphi \vee \psi)$, $(\varphi \Rightarrow \psi)$, and $\diamondsuit\!\!\!\!\diamond\varphi$ are introduced as the usual abbreviations for $\neg$**true**, $\neg(\neg\varphi \wedge \neg\psi)$, $(\neg\varphi \vee \psi)$, and $\neg\boxed{a}\neg\varphi$, respectively. Formulae are interpreted with respect to the distinguished world w_0, that is, satisfiability is defined with respect to pointed-models. A formula φ is said to be *satisfied in the model $\mathcal{M} = (\mathcal{S}, \pi)$ of the Kripke frame $\mathcal{S} = (\mathcal{W}, w_0, \mathcal{R}_1, \ldots, \mathcal{R}_n)$* if $\langle \mathcal{M}, w_0 \rangle \models \varphi$; the formula φ is *satisfiable in a Kripke frame $\mathcal{S}$* if there is a model $\mathcal{M}$ of $\mathcal{S}$ such that $\langle \mathcal{M}, w_0 \rangle \models \varphi$; and φ is said to be *valid in a class $\mathcal{C}$ of Kripke frames* if it is satisfied in any model of any Kripke frame belonging to the class $\mathcal{C}$.

3 Resolution for $\mathsf{K}_{(n)}$

In [17], a sound, complete, and terminating resolution-based method for $\mathsf{K}_{(n)}$, which in this paper we call $\mathsf{RES_K}$, is introduced. As the proof-method for logics of confluence presented here relies on $\mathsf{RES_K}$, in order to keep the present paper self-contained, we reproduce the corresponding inference rules here and refer the reader to [17] for a detailed account of the method. The approach taken in the resolution-based method for $\mathsf{K}_{(n)}$ is clausal: a formula to be tested for (un)satisfiability is first translated into a normal form, explained in Section 3.1, and then the inference rules given in Section 3.2 are applied until either a contradiction is found or no new clauses can be generated.

3.1 A Normal Form for $\mathsf{K}_{(n)}$

Formulae in the language of $\mathsf{K}_{(n)}$ can be transformed into a normal form called Separated Normal Form for Normal Logics (SNF). As the semantics is given with respect to a pointed-model, we add a nullary connective **start** in order to represent the world from which we start reasoning. Formally, given a model $\mathcal{M} = (\mathcal{W}, w_0, \mathcal{R}_1, \ldots, \mathcal{R}_n, \pi)$, we have that $\langle \mathcal{M}, w \rangle \models$ **start** if, and only if, $w = w_0$. A formula in SNF is represented by a conjunction of clauses, which are true at all reachable states, that is, they have the general form $\bigwedge_i \square^* A_i$, where A_i is a clause and $\square^*$, the universal operator, is characterised

by (the greatest fixed point of) $\Box^*\varphi \Leftrightarrow \varphi \wedge \bigwedge_{a \in \mathcal{A}_n} \boxed{a}\Box^*\varphi$, for a formula φ. Observe that satisfaction of $\Box^*\varphi$ imposes that φ must hold at the actual world w and at every world reachable from w, where reachability is defined in the usual (graph-theoretic) way. Clauses have one of the following forms:

- Initial clause $\qquad\qquad\qquad\qquad\qquad\qquad$ $\textbf{start} \Rightarrow \bigvee_{b=1}^{r} l_b$

- Literal clause $\qquad\qquad\qquad\qquad\qquad\qquad$ $\textbf{true} \Rightarrow \bigvee_{b=1}^{r} l_b$

- Positive a-clause $\qquad\qquad\qquad\qquad\qquad\quad$ $l' \Rightarrow \boxed{a}\, l$

- Negative a-clause $\qquad\qquad\qquad\qquad\qquad\quad$ $l' \Rightarrow \neg \boxed{a}\, l$

where l, l', $l_b \in \mathcal{L}$. 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.

The translation to SNF uses rewriting of classical operators and the renaming technique [20], where complex subformulae are replaced by new propositional symbols and the truth of these new symbols is linked to the formulae that they replaced in all states. Given a formula φ, the translation procedure is applied to $\Box^*(\textbf{start} \Rightarrow t_0) \wedge \Box^*(t_0 \Rightarrow \varphi)$, where t_0 is a new propositional symbol. The universal operator, which surrounds all clauses, ensures that the clauses generated by the translation of a formula are true at all reachable worlds. Classical rewriting is used to remove some classical operators from φ (e.g. $\Box^*(t \Rightarrow \psi_1 \wedge \psi_2)$ is rewritten as $\Box^*(t \Rightarrow \psi_1) \wedge \Box^*(t \Rightarrow \psi_2)$). Renaming is used to replace complex subformulae in disjunctions (e.g. if ψ_2 is not a literal, $\Box^*(t \Rightarrow \psi_1 \vee \psi_2)$ is rewritten as $\Box^*(t \Rightarrow \psi_1 \vee t_1) \wedge \Box^*(t_1 \Rightarrow \psi_2)$, where t_1 is a new propositional symbol) or in the scope of modal operators (e.g. if ψ is not a literal, $\Box^*(t \Rightarrow \boxed{a}\psi)$ is rewritten as $\Box^*(t \Rightarrow \boxed{a}t_1) \wedge \Box^*(t_1 \Rightarrow \psi)$, where t_1 is a new propositional symbol). We refer the reader to [17] for details on the transformation rules that define the translation to SNF, their correctness, and examples of their application.

3.2 Inference Rules for $\mathsf{K}_{(n)}$

In the following, l, l', l_i, $l_i' \in \mathcal{L}$ ($i \in \mathbb{N}$) and D, D' are disjunctions of literals.

Literal Resolution. This is classical resolution applied to the classical propositional fragment of the combined logic. An initial clause may be resolved with either a literal clause or another initial clause (rules **IRES1** and **IRES2**). Literal clauses may be resolved together (**LRES**).

$$[\textbf{IRES1}] \; \Box^*(\textbf{true} \Rightarrow D \vee l) \qquad [\textbf{IRES2}] \; \Box^*(\textbf{start} \Rightarrow D \vee l) \qquad [\textbf{LRES}] \; \Box^*(\textbf{true} \Rightarrow D \vee l)$$
$$\dfrac{\Box^*(\textbf{start} \Rightarrow D' \vee \neg l)}{\Box^*(\textbf{start} \Rightarrow D \vee D')} \qquad\quad \dfrac{\Box^*(\textbf{start} \Rightarrow D' \vee \neg l)}{\Box^*(\textbf{start} \Rightarrow D \vee D')} \qquad\quad \dfrac{\Box^*(\textbf{true} \Rightarrow D' \vee \neg l)}{\Box^*(\textbf{true} \Rightarrow D \vee D')}$$

Modal Resolution. These rules are applied between clauses which refer to the same context, that is, they must refer to the same agent. For instance, we may resolve two

or more $\boxed{a}$-clauses (rules **MRES** and **NEC2**); or several $\boxed{a}$-clauses and a literal clause (rules **NEC1** and **NEC3**). The modal inference rules are:

$$[\textbf{MRES}] \quad \Box^*(l_1 \Rightarrow \boxed{a}\, l)$$
$$\frac{\Box^*(l_2 \Rightarrow \neg\,\boxed{a}\, l)}{\Box^*(\textbf{true} \Rightarrow \neg l_1 \vee \neg l_2)}$$

$$[\textbf{NEC2}] \quad \Box^*(l_1' \Rightarrow \boxed{a}\, l_1)$$
$$\Box^*(l_2' \Rightarrow \boxed{a}\, \neg l_1)$$
$$\frac{\Box^*(l_3' \Rightarrow \neg\,\boxed{a}\, l_2)}{\Box^*(\textbf{true} \Rightarrow \neg l_1' \vee \neg l_2' \vee \neg l_3')}$$

$$[\textbf{NEC1}] \quad \Box^*(l_1' \Rightarrow \boxed{a}\, \neg l_1)$$
$$\vdots$$
$$\Box^*(l_m' \Rightarrow \boxed{a}\, \neg l_m)$$
$$\Box^*(l' \Rightarrow \neg\,\boxed{a}\, l)$$
$$\frac{\Box^*(\textbf{true} \Rightarrow l_1 \vee \ldots \vee l_m \vee l)}{\Box^*(\textbf{true} \Rightarrow \neg l_1' \vee \ldots \vee \neg l_m' \vee \neg l')}$$

$$[\textbf{NEC3}] \quad \Box^*(l_1' \Rightarrow \boxed{a}\, \neg l_1)$$
$$\vdots$$
$$\Box^*(l_m' \Rightarrow \boxed{a}\, \neg l_m)$$
$$\Box^*(l' \Rightarrow \neg\,\boxed{a}\, l)$$
$$\frac{\Box^*(\textbf{true} \Rightarrow l_1 \vee \ldots \vee l_m)}{\Box^*(\textbf{true} \Rightarrow \neg l_1' \vee \ldots \vee \neg l_m' \vee \neg l')}$$

The rule **MRES** is a syntactic variation of classical resolution, as a formula and its negation cannot be true at the same state. The rule **NEC1** corresponds to necessitation (applied to $(\neg l_1 \wedge \ldots \wedge \neg l_m \Rightarrow \neg l)$, which is equivalent to the literal clause in the premises) and several applications of classical resolution. The rule **NEC2** is a special case of **NEC1**, as the parent clauses can be resolved with the tautology $\textbf{true} \Rightarrow l_1 \vee \neg l_1 \vee l_2$. The rule **NEC3** is similar to **NEC1**, however the negative modal clause is not resolved with the literal clause in the premises. Instead, the negative modal clause requires that resolution takes place between literals on the right-hand side of positive modal clauses and the literal clause. The resolvents in the inference rules **NEC1**–**NEC3** impose that the literals on the left-hand side of the modal clauses in the premises are not all satisfied whenever their conjunction leads to a contradiction in a successor state. Given the syntactic forms of clauses, the three rules are needed for completeness [17]. Note that for **NEC1**, we may have $m = 0$; for **NEC2** the number of premises is fixed; and that for **NEC3**, if $m = 0$, then the literal clause in the premises is $\textbf{true} \Rightarrow \textbf{false}$, which cannot be satisfied in any model. Thus, **NEC3** is not applied when $m = 0$.

We define a derivation as a sequence of sets of clauses $\mathcal{T}_0, \mathcal{T}_1, \ldots$, where $\mathcal{T}_i$ results from adding to $\mathcal{T}_{i-1}$ the resolvent obtained by an application of an inference rule of $\mathsf{RES_K}$ to clauses in $\mathcal{T}_{i-1}$. A derivation *terminates* if, and only if, either a contradiction, in the form of $\Box^*(\textbf{start} \Rightarrow \textbf{false})$ or $\Box^*(\textbf{true} \Rightarrow \textbf{false})$, is derived or no new clauses can be derived by further application of the resolution rules of $\mathsf{RES_K}$. We assume standard simplification from classical logic to keep the clauses as simple as possible. For example, $D \vee l \vee l$ on the right-hand side of a clause would be rewritten as $D \vee l$.

Example 1. We wish to check whether the formula $\boxed{1}\,\boxed{2}(a \wedge b) \Rightarrow \boxed{1}(\boxed{2}a \wedge \boxed{2}b)$ is valid in $\mathsf{K}_{(2)}$. The translation of its negation into the normal form is given by clauses (1)–(9) below. Then the inference rules are applied until **false** is generated. In order to improve readability, the universal operator is suppressed. The full refutation follows.

1. **start** $\Rightarrow t_1$ 9. $t_6 \Rightarrow \neg\, \boxed{2}\, b$

2. $t_1 \Rightarrow \boxed{1}\, t_2$ 10. **true** $\Rightarrow \neg t_2 \vee \neg t_5$ [**NEC1**, $3, 8, 4$]

3. $t_2 \Rightarrow \boxed{2}\, t_3$ 11. **true** $\Rightarrow \neg t_2 \vee \neg t_4 \vee t_6$ [**LRES**, $10, 7$]

4. **true** $\Rightarrow \neg t_3 \vee a$ 12. **true** $\Rightarrow \neg t_2 \vee \neg t_6$ [**NEC1**, $3, 9, 5$]

5. **true** $\Rightarrow \neg t_3 \vee b$ 13. **true** $\Rightarrow \neg t_2 \vee \neg t_4$ [**LRES**, $12, 11$]

6. $t_1 \Rightarrow \neg\, \boxed{1}\, \neg t_4$ 14. **true** $\Rightarrow \neg t_1$ [**NEC1**, $2, 6, 13$]

7. **true** $\Rightarrow \neg t_4 \vee t_5 \vee t_6$ 15. **start** $\Rightarrow$ **false**   [**IRES1**, $14, 1$]

8. $t_5 \Rightarrow \neg\, \boxed{2}\, a$

Clauses (10) and (12) are obtained by applications of **NEC1** to clauses in the context of agent 2. Clause (14) is obtained by an application of the same rule, but in the context of agent 1. Clauses (11) and (13) result from applications of resolution to the propositional part of the language shared by both agents. Clause (15) shows that a contradiction was found at the initial state. Therefore, the original formula is valid.

4   Clausal Resolution for Logics of Confluence

The inference rules of $\mathsf{RES_K}$, given in Section 3.2, are resolution-based: whenever a set of (sub)formulae is identified as contradictory, the resolvents require that they are not all satisfied together. The extra inference rules for $\mathsf{K}^{p,q,r,s}_{(n)}$, with $p, q, r, s \in \{0, 1\}$, which we are about to present, have a different flavour: whenever we can identify that the set of clauses imply that $\diamondsuit^p\, \boxed{a}\,^q \psi$ holds, we add some new clauses that ensure that $\boxed{a}\,^r \diamondsuit^s \psi$ also holds. If this is not the case, that is, if the set of clauses implies that $\neg\, \boxed{a}\,^r \diamondsuit^s \psi$ holds, then a contradiction is found by applying the inference rules for $\mathsf{K}_{(n)}$. Because of the particular normal form we use here, there are, in fact, two general forms for the inference rules for $\mathsf{K}^{p,q,r,s}_{(n)}$, given in Table 2 (where l, l' are literals and C is a conjunction of literals).

Table 2. Inference Rules for $\mathbf{G}^{p,q,r,s}_a$

$[\mathbf{RES}^{p,1,r,s}_a] \quad \dfrac{\Box^*(l \Rightarrow \boxed{a}\, l')}{\Box^*(\diamondsuit^p l \Rightarrow \boxed{a}\,^r \diamondsuit^s l')}$	$[\mathbf{RES}^{p,0,r,s}_a] \dfrac{\Box^*(C \Rightarrow \diamondsuit^p l')}{\Box^*(C \Rightarrow \boxed{a}\,^r \diamondsuit^s l')}$

Soundness is checked by showing that the transformation of a formula $\varphi \in \mathsf{WFF}_{\mathsf{K}_{(n)}}$ into its normal form is satisfiability-preserving and that the application of the inference rules are also satisfiability-preserving. Satisfiability-preserving results for the transformation into SNF are provided in [17]. To extend the soundness results so as to cover the new inference rules, note that the conclusions of the inference rules in Table 2 are derived using the semantics of the universal operator and the distribution axiom, $\mathbf{K}_a$. For $\mathbf{RES}^{p,1,r,s}_a$, we have that the premise $\Box^*(l \Rightarrow \boxed{a}\, l')$ is semantically equivalent to $\Box^*(\neg\, \boxed{a}\, l' \Rightarrow \neg l)$. By the definition of the universal operator, we obtain $\Box^*(\boxed{a}\,^p(\neg\, \boxed{a}\, l' \Rightarrow \neg l))$. Applying the distribution axiom $\mathbf{K}_a$ to this clause results in $\Box^*(\boxed{a}\,^p \neg\, \boxed{a}\, l' \Rightarrow \boxed{a}\,^p \neg l))$, which is semantically equivalent to $\Box^*(\neg\, \boxed{a}\,^p \neg l \Rightarrow$

$\neg \boxed{a}^p \neg \boxed{a} l')$. As $\Diamond$ is an abbreviation for $\neg \boxed{a} \neg$ and because $\Diamond^p \boxed{a} l'$ implies $\boxed{a}^r \Diamond^s l'$ in $\mathsf{K}_{(n)}^{p,1,r,s}$, by classical reasoning, we have that $\Box^*(\neg \boxed{a}^p \neg l \Rightarrow \neg \boxed{a}^p \neg \boxed{a} l')$ implies $\Box^*(\Diamond^p l \Rightarrow \boxed{a}^r \Diamond^s l')$, the conclusion of $\mathbf{RES}_a^{p,1,r,s}$. Soundness of the inference rule $\mathbf{RES}_a^{p,0,r,s}$ can be proved in a similar way.

As the conclusions of the above inference rules may contain complex formulae, they might need to be rewritten into the normal form. Thus, we also need to add clauses corresponding to the normal form of $\Diamond^p l$ and $\Diamond^s l'$, which occur in the conclusions of the inference rules. Let φ be a formula and let $\tau(\varphi)$ be the set of clauses resulting from the translation of φ into the normal form. Let $\mathcal{L}(\tau(\varphi))$ be the set of literals that might occur in the clause set, that is, for all $p \in \mathcal{P}$ such that p occurs in $\tau(\varphi)$, we have that both p and $\neg p$ are in $\mathcal{L}(\tau(\varphi))$. The set of *definition clauses* is given by

$$\Box^*(pos_{a,l} \Rightarrow \neg \boxed{a} \neg l)$$

$$\Box^*(\neg pos_{a,l} \Rightarrow \boxed{a} \neg l)$$

for all $l \in \mathcal{L}(\tau(\varphi))$, where $pos_{a,l}$ is a new propositional *definition symbol* used for renaming the negative modal literal $\Diamond l$, that is, the definition clauses correspond to the normal form of $pos_{a,l} \Leftrightarrow \neg \boxed{a} \neg l$. Note that we have definition clauses for every propositional symbol and its negation, e.g. for a propositional symbol $p \in \tau(\varphi)$, we have the definition clauses $\Box^*(pos_{a,p} \Rightarrow \neg \boxed{a} \neg p)$, $\Box^*(\neg pos_{a,p} \Rightarrow \boxed{a} \neg p)$, $\Box^*(pos_{a,\neg p} \Rightarrow \neg \boxed{a} p)$, and $\Box^*(\neg pos_{a,\neg p} \Rightarrow \boxed{a} p)$, for every $a \in \mathcal{A}_n$ occurring in $\tau(\varphi)$. We assume the set of definition clauses to be available whenever those symbols are used. It is also important to note that those new definition symbols and the respective definition clauses can all be introduced at the beginning of the application of the resolution method because we do not need definition clauses applied to definition symbols in the proofs, as given in the completeness proof [18]. As no new propositional symbols are introduced by the inference rules, there is a finite number of clauses that might be expressed (modulo simplification) and, therefore, the clausal resolution method for each modal logic of confluence is terminating.

As discussed above and from the results in [17], we can establish the soundness of the proof method.

Theorem 1. *The resolution-based calculi for logics of confluence are sound.*

Proof (Sketch). The transformation into the normal form is satisfiability preserving [17]. Given a set $\mathcal{T}$ of clauses and a model $\mathcal{M}$ that satisfies $\mathcal{T}$, we can construct a model $\mathcal{M}'$ for the union of $\mathcal{T}$ and the definition clauses, where $\mathcal{M}$ and $\mathcal{M}'$ may differ only in the valuation of the definition symbols. By setting properly the valuations in $\mathcal{M}'$, we have that $\langle \mathcal{M}', w \rangle \models pos_{a,p}$ if and only if $\langle \mathcal{M}, w \rangle \models \Diamond p$, for any $w \in \mathcal{W}$. Soundness of the inference rules for RES_K is also given in [17]. Soundness of $\mathbf{RES}_a^{p,1,r,s}$ and $\mathbf{RES}_a^{p,0,r,s}$ follow from the axiomatisation of $\mathsf{K}_{(n)}^{p,q,r,s}$.

Table 3 shows the inference rules for each specific instance of $\mathbf{G}_a^{p,q,r,s}$, where $p, q, r, s \in \{0, 1\}$, l, $l' \in \mathcal{L}$, and D is a disjunction of literals. As $\mathbf{G}_a^{p,q,r,s}$ is semantically equivalent to $\mathbf{G}_a^{r,s,p,q}$, the inference rules for both systems are grouped together. Some of the inference rules in Table 3 are obtained directly from Table 2. For instance, the rule for reflexive systems, i.e. where the axiom $\mathbf{G}_a^{0,1,0,0}$ is valid, has the

Table 3. Inference Rules for several instances of $\mathbf{G}_a^{p,q,r,s}$

Logic	Inference Rules
$\mathbf{T}_a$	[$\mathbf{RES}_a^{0,0,0,1}$] $\dfrac{\Box^*(\mathbf{true} \Rightarrow D \vee l)}{\Box^*(\neg D \Rightarrow \neg \boxed{a} \neg l)}$ [$\mathbf{RES}_a^{0,1,0,0}$] $\dfrac{\Box^*(l \Rightarrow \boxed{a} l')}{\Box^*(\mathbf{true} \Rightarrow \neg l \vee l')}$
$\mathbf{Ban}_a$	[$\mathbf{RES}_a^{0,0,1,0}$] $\dfrac{\Box^*(\mathbf{true} \Rightarrow D \vee l)}{\Box^*(\neg D \Rightarrow \boxed{a} l)}$ [$\mathbf{RES}_a^{1,0,0,0}$] $\dfrac{\Box^*(l \Rightarrow \neg \boxed{a} \neg l')}{\Box^*(\mathbf{true} \Rightarrow \neg l \vee l')}$
$\mathbf{B}_a$	[$\mathbf{RES}_a^{0,0,1,1}$] $\dfrac{\Box^*(\mathbf{true} \Rightarrow D \vee l)}{\Box^*(\neg D \Rightarrow \boxed{a} pos_{a,l})}$ [$\mathbf{RES}_a^{1,1,0,0}$] $\dfrac{\Box^*(l \Rightarrow \boxed{a} l')}{\Box^*(\neg l' \Rightarrow \boxed{a} \neg l)}$
$\mathbf{D}_a$	[$\mathbf{RES}_a^{0,1,0,1}$] $\dfrac{\Box^*(l \Rightarrow \boxed{a} l')}{\Box^*(l \Rightarrow \neg \boxed{a} \neg l')}$

Logic	Inference Rules
$\mathbf{G}_a^{0,1,1,1}$	[$\mathbf{RES}_a^{0,1,1,1}$] $\dfrac{\Box^*(l \Rightarrow \boxed{a} l')}{\Box^*(l \Rightarrow \boxed{a} pos_{a,l'})}$ [$\mathbf{RES}_a^{1,0,1,1}$] $\dfrac{\Box^*(l \Rightarrow \boxed{a} l')}{\Box^*(pos_{a,l} \Rightarrow \neg \boxed{a} \neg l')}$
$\mathbf{F}_a$	[$\mathbf{RES}_a^{1,0,1,0}$] $\dfrac{\Box^*(l \Rightarrow \neg \boxed{a} \neg l')}{\Box^*(l \Rightarrow \boxed{a} l')}$
$\mathbf{5}_a$	[$\mathbf{RES}_a^{1,0,1,1}$] $\dfrac{\Box^*(l \Rightarrow \neg \boxed{a} \neg l')}{\Box^*(l \Rightarrow \boxed{a} pos_{a,l'})}$ [$\mathbf{RES}_a^{1,1,1,0}$] $\dfrac{\Box^*(l \Rightarrow \boxed{a} l')}{\Box^*(pos_{a,l} \Rightarrow \boxed{a} l')}$
$\mathbf{G1}_a$	[$\mathbf{RES}_a^{1,1,1,1}$] $\dfrac{\Box^*(l \Rightarrow \boxed{a} l')}{\Box^*(pos_{a,l} \Rightarrow \boxed{a} pos_{a,l'})}$

form $\Box^*(l \Rightarrow \boxed{a} l')/\Box^*(\langle\!\langle a\rangle\!\rangle^0 l \Rightarrow \boxed{a}^0 \langle\!\langle a\rangle\!\rangle^0 l')$ in Table 2; in Table 3, the conclusion is rewritten in its normal form, that is, $\Box^*(\mathbf{true} \Rightarrow \neg l \vee l')$. For other systems, the form of the inference rules are slightly different from what would be obtained from a direct application of the general inference rules in Table 2. This is the case, for instance, for the inference rules for symmetric systems, that is, those systems where the axiom $\mathbf{G}_a^{1,1,0,0}$ is valid. From Table 2, in symmetric systems, for a premise of the form $\Box^*(l \Rightarrow \boxed{a} l')$, the conclusion is given by $\Box^*(\langle\!\langle a\rangle\!\rangle l \Rightarrow l')$, which is translated into the normal form as $\Box^*(\mathbf{true} \Rightarrow \neg pos_{a,l} \vee l')$. We have chosen, however, to translate the conclusion as $\Box^*(\neg l' \Rightarrow \boxed{a} \neg l)$, which is semantically equivalent to the conclusion obtained by the general inference rule, but avoids the use of definition symbols.

The inference rules given in Table 2 provide a systematic way of designing the inference rules for each specific modal logic of confluence. We note, however, that we do not always need both inference rules in order to achieve a complete proof method for a particular logic. In the completeness proofs provided in [18], we show for instance that the inference rules which introduce modalities in their conclusions from literal clauses (that is, the inference rules $\mathbf{RES}_a^{0,0,r,s}$) are not needed for completeness. We also show that we need just one specific inference rule for logics in which $\mathbf{G}_a^{0,1,1,1}$ and $\mathbf{5}_a$ are valid: $\mathbf{RES}_a^{0,1,1,1}$ and $\mathbf{RES}_a^{1,0,1,1}$, respectively.

Given a formula φ in $\mathsf{K}_{(n)}^{p,q,r,s}$, with $p,q,r,s \in \{0,1\}$, the resolution method for $\mathsf{K}_{(n)}$, given in Section 3, and the inference rule $\mathbf{RES}_a^{p,q,r,s}$ are applied to $\tau(\varphi)$ and the set of definition clauses. The extra inference rules for $\mathsf{K}_{(n)}^{p,q,r,s}$ do not need to be applied to clauses if such application generates new nested definition symbols, that is, we do not need definition clauses for definition symbols. For instance,

the application of $\mathbf{RES}_a^{1,1,1,1}$ to a clause of the form $\square^*(l \Rightarrow \boxed{a}\,pos_{a,l'})$ would result in $\square^*(pos_{a,l} \Rightarrow \boxed{a}\,pos_{a,pos_{a,l'}})$. Although it is not incorrect to apply the inference rules to such a clause, this might cause the method not to terminate. We can show, however, that the application of inference rules to clauses which would result in nested literals is not needed for completeness, as the restrictions imposed by those symbols are already ensured by existing definition symbols and relevant inference rules (see Theorem 3 below). This ensures that no new definition symbols are introduced by the proof method.

Completeness is proved by showing that, for each specific logic of confluence, if a given set of clauses 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 a set of clauses. The graph construction is similar to the construction of a canonical model, followed by filtrations based on the set of formulae (or clauses), often used to check completeness for proof methods in modal logics (see [3], for instance, for definitions and examples). Intuitively, nodes in the graph correspond to states and are defined as maximally consistent sets of literals and modal literals occurring in the set of clauses, including those literals introduced by definition clauses. That is, for any literal l occurring in the set of clauses, including definition clauses, and agents $a \in \mathcal{A}_n$, a node contains either l or $\neg l$; and either $\boxed{a}\,l$ or $\neg\,\boxed{a}\,l$. The set of edges correspond to the agents' accessibility relations. Edges or nodes that do not satisfy the set of clauses are deleted from the graph. Such deletions correspond to applications of one or more of the inference rules. We prove that an empty behaviour graph corresponds to an unsatisfiable set of clauses and that, in this case, there is a refutation using the inference rules for $\mathsf{RES_K}$, given in Section 3, and the inference rules for the specific logic of confluence, presented in Table 3.

Theorem 2. *Let $\mathcal{T}$ be an unsatisfiable set of clauses in $\mathbf{G}_a^{p,q,r,s}$, with $p, q, r, s \in \{0, 1\}$. A contradiction can be derived by applying the resolution rules for $\mathsf{RES_K}$, presented in Section 3, and Table 3.*

Proof (Sketch). We construct a behaviour graph and show that the application of rules in Table 3 removes nodes and edges where the corresponding frame condition does not hold. The full proof is provided in [18].

Theorem 3. *The resolution-based calculi for logics of confluence terminate.*

Proof (Sketch). From the completeness proof, the introduction of a literal such as $pos_{a,pos_{a,l}}$ for an agent a and literal l is not needed. We can show that the restrictions imposed by such clauses, together with the resolution rules for each specific logical system, are enough to ensure that the corresponding frame condition already holds. As the proof method does not introduce new literals in the clause set, there is only a finite number of clauses that can be expressed. Therefore, the proof method is terminating.

Example 2. We show that $\varphi \stackrel{\text{def}}{=} p \Rightarrow \square \Diamond p$, which is an instance of $\mathbf{B}_1$, is a valid formula in symmetric systems. As symmetry is implied by reflexivity and Euclideanness, instead of using $\mathbf{RES}_1^{1,1,0,0}$, we combine the inference rules for both $\mathbf{T}_1$ and $\mathbf{5}_1$. Clauses (1)–(4) correspond to the translation of the negation of φ into the normal form. Clauses (5)–(8) are the definition clauses used in the proof.

1.	$\mathbf{start} \Rightarrow t_0$		9.	$\mathbf{true} \Rightarrow \neg t_0 \vee pos_{1,t_1}$	$[\mathbf{MRES}, 5, 3]$
2.	$\mathbf{true} \Rightarrow \neg t_0 \vee p$		10.	$\mathbf{true} \Rightarrow \neg t_1 \vee \neg pos_{1,p}$	$[\mathbf{MRES}, 7, 4]$
3.	$t_0 \Rightarrow \neg \boxed{1} \neg t_1$		11.	$pos_{1,p} \Rightarrow \boxed{1} pos_{1,p}$	$[\mathbf{RES}_1^{1,0,1,1}, 7]$
4.	$t_1 \Rightarrow \boxed{1} \neg p$		12.	$\mathbf{true} \Rightarrow \neg pos_{1,p} \vee \neg pos_{1,t_1}$	$[\mathbf{NEC1}, 11, 6, 10]$
5.	$\neg pos_{1,t_1} \Rightarrow \boxed{1} \neg t_1$	$[\mathbf{Def.}\ pos_{1,t_1}]$	13.	$\mathbf{true} \Rightarrow \neg p \vee pos_{1,p}$	$[\mathbf{RES}_1^{0,1,0,0}, 8]$
6.	$pos_{1,t_1} \Rightarrow \neg \boxed{1} \neg t_1$	$[\mathbf{Def.}\ pos_{1,t_1}]$	14.	$\mathbf{true} \Rightarrow \neg p \vee \neg pos_{1,t_1}$	$[\mathbf{LRES}, 13, 12]$
7.	$pos_{1,p} \Rightarrow \neg \boxed{1} \neg p$	$[\mathbf{Def.}\ pos_{1,p}]$	15.	$\mathbf{true} \Rightarrow \neg t_0 \vee \neg p$	$[\mathbf{LRES}, 14, 9]$
8.	$\neg pos_{1,p} \Rightarrow \boxed{1} \neg p$	$[\mathbf{Def.}\ pos_{1,p}]$	16.	$\mathbf{true} \Rightarrow \neg t_0$	$[\mathbf{LRES}, 15, 2]$
			17.	$\mathbf{start} \Rightarrow \mathbf{false}$	$[\mathbf{IRES1}, 16, 1]$

Clause (11) results from applying the Euclidean inference rule to clause (7). Clause (13) results from applying the reflexive inference rule to (8). The remaining clauses are derived by the resolution calculus for $\mathsf{K}_{(1)}$. As a contradiction is found, given by clause (17), the set of clauses is unsatisfiable and the original formula φ is valid.

5   Closing Remarks

We have presented a sound, complete, and terminating proof method for logics of confluence, that is, normal multimodal systems where axioms of the form

$$\mathbf{G}_a^{p,q,r,s} = \Diamond^p \boxed{a}^q \varphi \Rightarrow \boxed{a}^r \Diamond^s \varphi$$

where $p, q, r, s \in \{0, 1\}$, are valid. The axioms $\mathbf{G}_a^{p,q,r,s}$ provide a general form for axioms widely used in logical formalisms applied to representation and reasoning within Computer Science.

We have proved completeness of the proof method presented in this paper for eight families of logics and their fusions. The inference rules for particular instances of these logics can be systematically obtained and the resulting calculus can be implemented by adding to the existing prover for $\mathsf{K}_{(n)}$ [24] the clauses dependent on the clause-set. Efficiency, of course, depends on several aspects. Firstly, for certain classes of problems, dedicated proof methods might be more efficient. For instance, if the satisfiability problem for a particular logic is in NP (as in the case of $\mathsf{S5}_{(1)}$), then our procedure may be less efficient as the satisfiability problem for $\mathsf{K}_{(1)}$ is already PSPACE-complete [15]. Secondly, efficiency might depend on the inference rules chosen to produce proofs for a specific logic. For instance, for $\mathsf{S5}_{(n)}$, the user can choose the inference rules related to reflexivity and Euclideanness, or choose the inference rules related to seriality, symmetry, and Euclideanness. The number of inference rules used to test the unsatisfiability of a set of clauses for a particular logic might affect the number of clauses generated by the resolution method as well as the size of the proof. As in the case of derived inference rules in other proof methods, using more inference rules might lead to shorter proofs. Thirdly, as in the case of the resolution-based method for propositional logic, efficiency might be affected by strategies used to search for a proof. Future work includes the design of strategies for $\mathsf{RES}_\mathsf{K}(n)$ and for specific logics of confluence. Fourthly, efficiency might also depend on the form of the input problem. For instance, comparisons between tableaux methods and resolution methods [14,13] have shown that there is no

overall better approach: for some problems resolution proof methods behave better, for others tableaux based methods behave better. Providing a resolution-based method for the logics axiomatised by $\mathbf{K}_a$ and $\mathbf{G}_a^{p,q,r,s}$ gives the user a choice for automated tools that can be used depending on the type of the input formulae.

There are quite a few dedicated methods for the logics presented in this paper. In general, however, those methods do not provide a systematic way of dealing with logics based on similar axioms or their extensions. Therefore, we restrict attention here to methods related to logics of confluence. Tableaux methods for logics of confluence where the mono-modal axioms **T**, **D**, **B**, **4**, **5**, **De** (for density, the converse of **4**), and **G** are valid, can be found in [7,9]. For each of those axioms, a tableau inference rule is given. The inference rules can then be combined in order to provide proof methods for modal logics under $\mathsf{S5}_{(1)}$. Whilst the tableaux procedures in [7,9] are designed for mono-modal logics they seem to be extendable to multimodal logics as long as there are no interactions between modalities. Those procedures do not cover all the logics investigated in this paper. In [2], labelled tableaux are given for the mono-modal logics axiomatised by **K** and axioms $\mathbf{G}^{p,q,r,s}$ where $q = s = 0$ implies $p = r = 0$. This restriction avoids the introduction of inference rules related to the identity predicate, but also excludes, for instance, functional and modally banal systems, which are treated by the method introduced in the present paper. In [4], hybrid logic tableaux methods for logics of confluence are given: the inference rules create nodes, labelled by nominals. The nominals are used in order to eliminate the Skolem function related to the existential quantifier in the first-order sentence corresponding to the axiom $\mathbf{G}_a^{p,q,r,s}$. This proof method provides tableau rules for all instances of the axiom. Soundness and completeness are discussed, but termination of the method is not dealt with and it is not clear what are the bounds for creating new nodes in the general case. In [12], sound, complete, and terminating display calculi for tense logics and some of its extensions, including those with the axiom $\mathbf{G}_a^{p,q,r,s}$, are presented. It has been shown that these calculi have the property of *separation*, that is, they provide complete proof methods for the component fragments. The paper investigates the relation between the display calculi and deep inference systems (where the sequent rules can be applied at any node of a proof tree). By finding appropriate propagation rules for the fusion of tense logic with either $\mathsf{S4}_{(1)}$, $\mathsf{S5}_{(1)}$, or functional systems, completeness of search strategies are presented. However, propagation rules for the axiom of convergence, $\mathbf{G1}$, or for the combination of path axioms (i.e. axioms of the form $\Diamond^i \varphi \Rightarrow \Diamond^j \varphi$) with seriality are not given. Also related, in [1], prefixed tableaux procedures for confluence logics that validate the multimodal version of the axiom $\Diamond \boxed{b} \varphi \Rightarrow \boxed{c} \Diamond \varphi$, where φ is a formula, are given. Note that the logics in [1] are systems with instances of the axiom $\mathbf{G}_{a,b,c,d}^{1,1,1,1}$, that is, a logic which allows the interaction of the agents $a, b, c, d \in \mathcal{A}_n$, and might lead to undecidable systems.

To the best of our knowledge, there are no resolution-based proof methods for logics of confluence. However, resolution-based methods for modal logics, based on translation into first-order logic, have been proposed for several modal logics. A survey on translation-based approaches for non-transitive modal logics (i.e. modal logics that do not include the axiom **4**) can be found in [19]. The translation-based approach has the clear advantage of being easily implemented, making use of well-established theorem-provers, and dealing with any logic that can be embedded into first-order, should it be

decidable or not. However, first-order provers cannot deal easily with logics that embed some properties which are covered by particular axioms of confluence (e.g. functionality). In order to avoid such problematic fragments within first-order logic, the axiomatic translation principle for modal logic, introduced in [23], besides using the standard translation of a modal formulae into first-order, takes an axiomatisation for a particular modal logic and introduces a set of first-order modal axioms in the form of *schema clauses*. As an example, adapted from [23], in order to prove that $\boxed{a}\neg\boxed{a}p$ is satisfiable in $\mathsf{KT4}_{(n)}$, for each modal subformula (i.e. $\boxed{a}\neg\boxed{a}p$ and $\boxed{a}p$) and for each considered axiom (i.e. **T** and **4**), one schema clause is added, resulting in:

$$\neg Q_{\boxed{a}\neg\boxed{a}p}(x) \vee \neg R(x,y) \vee Q_{\boxed{a}\neg\boxed{a}p}(y) \qquad \neg Q_{\boxed{a}\neg\boxed{a}p}(x) \vee Q_{\neg\boxed{a}p}(y)$$

$$\neg Q_{\boxed{a}p}(x) \vee \neg R(x,y) \vee Q_{\boxed{a}p}(y) \qquad \neg Q_{\boxed{a}p}(x) \vee Q_p(y)$$

where the predicate $Q_\varphi(x)$ can be read as φ holds at world x and R is the predicate symbol to express the accessibility relation for agent a. Note that the clauses on the left are related to transitivity (**4**) and the two clauses on the right are related to reflexivity (**T**). The axiomatic translation approach is similar to the approach taken in the present paper and in [17] as the schema clauses provide a way of talking about properties of the accessibility relation. As in our case, soundness follows easily from the properties of the translation. Termination follows from the fact that only a finite number of schema clauses are needed. However, as in the case of the proof method presented here, general completeness of the method is difficult to be proved and it is given only for particular families of logics. In [10], a translation-based approach for properties which can be expressed by regular grammar logics (including transitivity and Euclideanness) is given. Completeness of the method has been proved for some families of logics.

In the present paper, we have restricted attention to the case where $p, q, r, s \in \{0, 1\}$, but we believe that the proof method can be extended in a uniform way for dealing with the unsatisfiability problem for any values of $p, q, r,$ and s, by adding inference rules of the following form:

$$[\mathbf{RES}_a^{p,q,r,s}] \frac{\Box^*(l \Rightarrow \Diamond^p \boxed{a}^r l')}{\Box^*(l \Rightarrow \boxed{a}^r \Diamond^s l')}$$

which requires search for clauses that correspond to the normal form of the premise and the introduction of as many new definition symbols as the number of modalities occurring in the conclusion. The inference rule $\mathbf{RES}_a^{p,q,r,s}$ is obviously sound, but we have yet to identify the restrictions on the number of new propositional symbols introduced by the method in order to ensure termination. Future work includes this extension, the complexity analysis, the implementation of the proof method, and practical comparisons with other methods.

References

1. Baldoni, M., Giordano, L., Martelli, A.: A tableau calculus for multimodal logics and some (un)decidability results. In: de Swart, H. (ed.) TABLEAUX 1998. LNCS (LNAI), vol. 1397, pp. 44–59. Springer, Heidelberg (1998)

2. Basin, D., Matthews, S., Viganò, L.: Labelled propositional modal logics: Theory and practice. J. Log. Comput 7(6), 685–717 (1997)
3. Blackburn, P., de Rijke, M., Venema, Y.: Modal Logic. Cambridge University Press, Cambridge (2001)
4. Blackburn, P., Dialogue, E.L.E., Cate, B.T.: Beyond pure axioms: Node creating rules in hybrid tableaux. In: Areces, C.E., Blackburn, P., Marx, M., Sattler, U. (eds.) Hybrid Logics, pp. 21–35 (July 25, 2002)
5. Boolos, G.S.: The Logic of Provability. Cambridge University Press (1993)
6. Carnielli, W.A., Pizzi, C.: Modalities and Multimodalities. Logic, Epistemology, and the Unity of Science, vol. 12. Springer (2008)
7. Castilho, M.A., del Cerro, L.F., Gasquet, O., Herzig, A.: Modal tableaux with propagation rules and structural rules. Fundamenta Informaticae 32(3-4), 281–297 (1997)
8. Chellas, B.: Modal Logic: An Introduction. Cambridge University Press (1980)
9. del Cerro, L.F., Gasquet, O.: Tableaux based decision procedures for modal logics of confluence and density. Fundamenta Informaticae 40(4), 317–333 (1999)
10. Demri, S., Nivelle, H.: Deciding regular grammar logics with converse through first-order logic. Journal of Logic, Language and Information 14(3), 289–329 (2005)
11. Fagin, R., Halpern, J., Moses, Y., Vardi, M.: Reasoning About Knowledge. MIT Press (1995)
12. Goré, R., Postniece, L., Tiu, A.: On the correspondence between display postulates and deep inference in nested sequent calculi for tense logics. Logical Methods in Computer Science 7(2) (2011)
13. Goré, R., Thomson, J., Widmann, F.: An experimental comparison of theorem provers for CTL. In: Combi, C., Leucker, M., Wolter, F. (eds.) TIME 2011, Lübeck, Germany, September 12-14, pp. 49–56. IEEE (2011)
14. Hustadt, U., Schmidt, R.A.: Scientific benchmarking with temporal logic decision procedures. In: Fensel, D., Giunchiglia, F., McGuinness, D., Williams, M.-A. (eds.) Proceedings of the KR 2002, pp. 533–544. Morgan Kaufmann (2002)
15. Ladner, R.E.: The computational complexity of provability in systems of modal propositional logic. SIAM J. Comput. 6(3), 467–480 (1977)
16. Lemmon, E.J., Scott, D.: The Lemmon Notes: An Introduction to Modal Logic. Segerberg, K. (ed.). Basil Blackwell (1977)
17. Nalon, C., Dixon, C.: Clausal resolution for normal modal logics. J. Algorithms 62, 117–134 (2007)
18. Nalon, C., Marcos, J., Dixon, C.: Clausal resolution for modal logics of confluence – extended version. Technical Report ULCS-14-001, University of Liverpool, Liverpool, UK (May 2014), `http://intranet.csc.liv.ac.uk/research/techreports/?id=ULCS-14-001`
19. Nivelle, H.D., Schmidt, R.A., Hustadt, U.: Resolution-based methods for modal logics. Logic J. IGPL 8 (2000)
20. Plaisted, D.A., Greenbaum, S.A.: A Structure-Preserving Clause Form Translation. Journal of Logic and Computation 2, 293–304 (1986)
21. Rao, A., Georgeff, M.: Modeling Rational Agents within a BDI-Architecture. In: Fikes, R., Sandewall, E. (eds.) Proceedings of KR&R-91, pp. 473–484. Morgan-Kaufmann (April 1991)
22. Schild, K.: A Correspondence Theory for Terminological Logics. In: Proceedings of the 12th IJCAI, pp. 466–471 (1991)
23. Schmidt, R.A., Hustadt, U.: The axiomatic translation principle for modal logic. ACM Transactions on Computational Logic 8(4), 1–55 (2007)
24. Silva, G.B.: Implementação de um provador de teoremas por resolução para lógicas modais normais. Monografia de Conclusão de Curso, Bacharelado em Ciência da Computação, Universidade de Brasília (2013), `http://www.cic.unb.br/~nalon/#software`