Journal of Automated Reasoning manuscript No.
(will be inserted by the editor)

Resolution with Order and Selection for Hybrid Logics

Carlos Areces · Daniel Gorín

4/06/2008

Abstract We investigate labeled resolution calculi for hybrid logics with inference rules restricted via selection functions and orders. We start by providing a sound and refutationally complete calculus for the hybrid logic $\mathscr{H}(@,\downarrow,\mathsf{A})$, even under restrictions by selection functions and orders. Then, by imposing further restrictions in the original calculus, we develop a sound, complete and terminating calculus for the $\mathscr{H}(@)$ sublanguage. The proof scheme we use to show refutational completeness of these calculi is an adaptation of a standard completeness proof for saturation-based calculi for first-order logic that guarantees completeness even under redundancy elimination. In fact, one of the contributions of this article is to show that the general framework of saturation-based proving for first-order logic with equality can be naturally adapted to saturation-based calculi for other languages, in particular modal and hybrid logics.

1 Introduction

In this article we study resolution based inference methods for hybrid logics. Our main, concrete contribution is to show that a labeled resolution calculus for the logic $\mathscr{H}(@,\downarrow,\mathsf{A})$ originally presented in [7] can be modified so that inference rules can be restricted via selection functions and orders [14]. More generally and, in our opinion, more fundamentally, the article shows how standard techniques from the well-established theory of first-order saturation-based reasoning can be adapted to this and similar calculi. This opens the way to efficient implementations, on the one hand, and further transfer of results and techniques on the other. We also show that a refinement of this calculus can be used as a decision method for the decidable fragment $\mathscr{H}(@)$.

State-of-the-art saturation-based methods for first-order logic have a very mature theory. The general framework presented by Bachmair and Ganzinger [14] can be used to establish the completeness of diverse calculi allowing their inference rules to be restricted with selection functions and orders. Carefully tailored selection functions and orders can then

Carlos Areces
Talaris Group, INRIA Nancy Grand Est, France. E-mail: areces@loria.fr

Daniel Gorín
Depto. de Computación, FCEyN, Universidad de Buenos Aires, Argentina. E-mail: dgorin@dc.uba.ar

be used to implement special refutation strategies, guarantee termination for certain fragments, or simply reduce the search space. Moreover, using this framework one can prove the completeness of a saturation-based method even in the presence of simplification rules and redundancy elimination. All these techniques play a crucial role in the implementation of efficient first-order automated theorem provers like Vampire [45], SPASS [53], etc.

For many modal and hybrid logics it is possible to define polynomial (sometimes even linear) time, satisfiability preserving translations into first-order logic. This has been a very successful approach and it takes advantage, in a natural way, of the well developed theorem proving methods for first-order logic. It originated with the work of Ohlbach (cf. [44]), and continued with the definition of different specialized translations into first-order logic which ensured better computational behavior and, in some cases, even termination of varied inference methods when applied to the output of the translation [48,49,8,34,50,36]. The translation based approach has even been proposed as a meta-framework for developing various calculi for different modal logics, as presented by Schmidt in [3]. This last article is, perhaps, the closest in spirit to the results we will present in this article. While Schmidt shows how tableau and other calculi can be developed for modal logics by using a suitable translation to first-order logic, and then applying the first-order resolution framework to synthesize inference rules, we aim to adapt the general framework of saturation-based inference of Bachmair and Ganzinger to calculi for logics other than first-order.

The saturation-based inference framework of Bachmair and Ganzinger was designed with first-order logic in mind (and targeted saturation-based calculi like resolution, superposition, etc.) but its definition is very abstract. As we are going to show in this article, it is indeed abstract enough to accommodate a labeled resolution calculus for hybrid logics.

We will present Bachmair and Ganzinger framework in detail in Section 3, but its abstract nature is already hinted by the following quote from [14, p. 35, no emphasis added]:

> We will next describe a comprehensive framework for modeling the key aspects of theorem proving, such as deduction, deletion, and simplification [. . .] The concepts and results in this chapter do not depend on details pertaining to specific syntax and representation of formulas, but apply to *general* clauses containing arbitrary quantifier-free subformulas [. . .] We assume that deduction is based on a clausal inference system Γ.

While Bachmair and Ganzinger were probably aiming to cover different calculi for diverse presentations and fragments of*first-order logic*[1], we will show that with suitable modifications the framework can be used on calculi for other languages (in our case, for hybrid logics). To the best of our knowledge, this is the first time the framework is used in this way.

The calculi we will consider target the very expressive hybrid language $\mathscr{H}(@,\downarrow,\mathsf{A})$ (which is expressively equivalent to the first-order language over the appropriate signature [10]). Interestingly, complete calculi (and, in some cases, even terminating calculi) for sublanguages of the full language can be obtained by eliminating/restricting some rules and defining suitable orders and selection functions. In Section 6 we show how this is done for the logic $\mathscr{H}(@)$. We conjecture that the general framework we present here can be used to develop complete and terminating calculi for other decidable fragments of $\mathscr{H}(@,\downarrow,\mathsf{A})$ (e.g., the extension of $\mathscr{H}(@)$ with converse operators, transitive modalities, etc.).

This article is organized as follows. In Section 2 we start with a brief introduction to modal and hybrid languages and discuss previous work on resolution in these languages,

[1] E.g., the restriction to *quantifier-free* formulas in the quote above is needed in the first-order case because of the particular way in which unification is handled; the calculi we are going to define work on ground formulas and do not require unification.

including a presentation of the labeled resolution calculus of [7]. In Section 3 we give an abstract description of the framework of Bachmair and Ganzinger (based on the presentation in [14]) and provide the outline of the general proof of refutational completeness for saturation-based proving, to set up the ground for the rest of the paper. In particular, we discuss the main differences between the first-order and the hybrid case. In Section 4 we prove the first main result in the paper (Theorem 1): the labeled resolution calculus $\mathbf{R}^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ is refutationally complete. This result, though, does not imply termination even for the decidable fragment $\mathscr{H}(@)$ of $\mathscr{H}(@,\downarrow,\mathsf{A})$. In Section 5 we show that we can change the paramodulation rule used in $\mathbf{R}^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ by a much more restricted rule while preserving refutational completeness. Moreover as we show in Section 6, using the restricted paramodulation rule, a terminating calculus for $\mathscr{H}(@)$ can be obtained, and hence a decision method (Theorem 7). In Section 7 we discuss the results presented in the paper. Detailed proofs have been moved to the Appendix.

2 Modal and Hybrid Logics

Modal logics [17, 18] are languages which were originally introduced to analyze concepts like necessity, possibility, etc [39]. Nowadays, they can be considered as languages which can capture fragments of first and higher-order logics and encapsulate them in a seemingly propositional set up. Their semantics, though, is presented in terms of relational first-order models (usually called Kripke models). Actually, we can obtain a quite accurate intuition of modal languages by thinking about them as languages specially devised to describe labeled graphs. Under this interpretation, the standard modal operators $\Box$ (which traditionally represented "necessity") and $\Diamond$ (which traditionally stood for "possibility") mean "in all successors of a particular state" and "in some successor of a particular state," respectively.

Modal languages offer relatively high expressive power, but unlike full classical first-order logic, they usually have a decidable satisfiability problem. For example, deciding satisfiability for the basic modal logic is PSpace-complete.

The classical modal language (i.e., the language containing only the relational modalities $\Box$ and $\Diamond$) suffers from some expressive limitations. On the one hand, it cannot make explicit reference to concrete elements of the domain (in terms of first-order logic, we would say that its signature does not contain constants). On the other hand, it can't express equality between elements. Hybrid logics [16, 10] are extensions of the classical modal logic that aim to solve these limitations by the introduction of nominals and special modal operators.

Intuitively, a nominal is a *name* for an element of a model even though, syntactically, it behaves exactly like a proposition symbol and can be used wherever the latter is acceptable. Nominals are simply a new sort of atomic symbols disjoint from the set of propositional variables. For example, if i is a nominal and p and q are propositional variables, then

$$\Box i \to (\Diamond q \to \Box q) \text{ and } \Box p \to (\Diamond q \to \Box q)$$

are both well formed hybrid formulas; but they have quite a different meaning. Actually, as we will now explain, the first is a tautology while the second is contingent. The difference comes from the interpretation that should be attributed to nominals. Because they are standing for particular elements in the model they should be true at a unique state. Formally, their interpretation is an element of the domain. Coming back to the examples above then, the antecedent $\Box i$ implies that the number of accessible states is at most one, and this condition is sufficient to make the consequent true (if q is true at some successor of the current state then it is true at all successors, given that the current state has at most one successor). This is, of course, not the case for the antecedent $\Box p$ of the second formula, as nothing forbids the existence of various accessible states where p holds.

But nominals are only the tip of the iceberg. Once we have names for states at our disposal we can introduce, for each nominal i, an operator $@_i$ that allows us to jump to the point named by i. The formula $@_i\varphi$ (read 'at i, φ') moves the point of evaluation to the state named by i and evaluates φ there. Intuitively, the $@_i$ operators internalize the satisfaction relation '$\models$' into the logical language:

$$M, w \models \varphi \text{ iff } M \models @_i\varphi, \text{ where } i \text{ is a nominal naming } w.$$

For this reason, these operators are usually called *satisfaction operators*.

Nominals and $@_i$ constitute the basic hybrid operators. The more expressive hybrid languages that we are going to investigate include the universal modality A and the 'bind-to-the-current-point' binder $\downarrow$. But let's introduce the language formally.

Definition 1 (Syntax) The set of formulas of the hybrid logic $\mathscr{H}(@, \downarrow, \mathsf{A})$ is defined with respect to a *signature* $\mathscr{S} = \langle \mathsf{PROP}, \mathsf{NOM}, \mathsf{REL}\rangle$, where $\mathsf{PROP} = \{p, q, r, \ldots\}$ (the proposition symbols), $\mathsf{NOM} = \{i, j, k, \ldots\}$ (the nominals) and $\mathsf{REL} = \{r_1, r_2, r_3, \ldots\}$ (the relation symbols) are infinite, enumerable, pairwise disjoint sets. $\mathsf{ATOM} = \mathsf{PROP} \cup \mathsf{NOM}$ is the set of atomic symbols. Given a signature $\mathscr{S}$ the set of $\mathscr{H}(@, \downarrow, \mathsf{A})$-formulas over $\mathscr{S}$ is defined as follows:

$$\mathscr{H}(@, \downarrow, \mathsf{A}) ::= a \mid \neg\varphi \mid \varphi \wedge \varphi' \mid [r]\varphi \mid @_i\varphi \mid {\downarrow} i.\varphi \mid \mathsf{A}\varphi$$

where $a \in \mathsf{ATOM}$, $i \in \mathsf{NOM}$, $r \in \mathsf{REL}$ and $\varphi, \varphi' \in \mathscr{H}(@, \downarrow, \mathsf{A})$. The remaining standard operators ($\langle r\rangle$, E, $\vee$, etc.) are defined in the usual way (i.e., $\langle r\rangle\varphi \equiv \neg[r]\neg\varphi$, $\mathsf{E}\varphi \equiv \neg\mathsf{A}\neg\varphi$, $\varphi \vee \varphi' \equiv \neg(\neg\varphi \wedge \neg\varphi')$).

An occurrence of a nominal i in a formula is *bound* whenever it is under the scope of a ${\downarrow}i$ operator. Non-bound occurrences of a nominal are called*free*. To avoid having to distinguish between free and bound occurrences of a given nominal in a formula we stipulate, without loss of generality, that no nominal appears both free and bound in a formula.

Except when indicated, in the rest of the paper we will assume a fixed arbitrary signature $\mathscr{S} = \langle \mathsf{PROP}, \mathsf{NOM}, \mathsf{REL}\rangle$.

Definition 2 (Semantics) A *hybrid model* is defined as a tuple $M = \langle W, (r^M)_{r \in \mathsf{REL}}, V, g\rangle$ where W is a non-empty set, $r^M \subseteq W \times W$ is a binary relation for each $r \in \mathsf{REL}$, and V and g are mappings such that $V(p) \subseteq W$ for each $p \in \mathsf{PROP}$, and $g(i) \in W$ for each $i \in \mathsf{NOM}$. Given a model $M = \langle W, (r^M)_{r \in \mathsf{REL}}, V, g\rangle$, an element $w \in W$ and a nominal i, we define the model M_i^w as $\langle W, (r^M)_{r \in \mathsf{REL}}, V, g'\rangle$ where g' is identical to g except perhaps in that $g'(i) = w$.

Given a hybrid model $M = \langle W, (r^M)_{r \in \mathsf{REL}}, V, g\rangle$ and an element $w \in W$ the satisfiability relation $M, w \models \varphi$ (read "model M satisfies formula φ at state w") is defined as follows:

$$\begin{array}{ll}
M, w \models p & \text{iff } w \in V(p),\ p \in \mathsf{PROP} \\
M, w \models i & \text{iff } w = g(i),\ i \in \mathsf{NOM} \\
M, w \models \neg\varphi & \text{iff } M, w \not\models \varphi \\
M, w \models \varphi_1 \wedge \varphi_2 & \text{iff } M, w \models \varphi_1 \text{ and } M, w \models \varphi_2 \\
M, w \models [r]\varphi & \text{iff } r^M(w, w') \text{ implies } M, w' \models \varphi, \text{ for all } w' \in W \\
M, w \models @_i\varphi & \text{iff } M, g(i) \models \varphi \\
M, w \models {\downarrow} i.\varphi & \text{iff } M_i^w, w \models \varphi \\
M, w \models \mathsf{A}\varphi & \text{iff } M, w' \models \varphi, \text{ for all } w' \in W.
\end{array}$$

We write $M \models \varphi$ if $M, w \models \varphi$ for all w in the domain, and if Φ is a set of formulas we write $M \models \Phi$ if $M \models \varphi$ for all $\varphi \in \Phi$.

The language $\mathscr{H}(@,\downarrow,\mathsf{A})$ is actually equivalent to first-order logic with equality over a vocabulary consisting of unary and binary predicate symbols, constants and no function symbols [10]. First-order quantification can be expressed in terms of $\downarrow$ and A: for example $\forall x.R(x,x)$ can be translated as $\mathsf{A}{\downarrow}x.\langle r\rangle x$. Intuitively, $\mathscr{H}(@,\downarrow,\mathsf{A})$ is decoupling the *quantificational* aspect (using A) from the *binding* aspect (using $\downarrow$) of the classical $\forall$ operator.

Interestingly, hybrid operators can be thought of in a *modular* way, giving rise to different subsystems of $\mathscr{H}(@,\downarrow,\mathsf{A})$, such as $\mathscr{H}(@)$, $\mathscr{H}(\mathsf{A})$, or $\mathscr{H}(@,\downarrow)$. Already the weakest hybrid logic we are going to consider, $\mathscr{H}(@)$, is more expressive than classical modal logic [4]. In particular it introduces, through nominals and @, a weak notion of equality reasoning. For example, the formulas

$$\begin{array}{ll}
@_i i & \text{(reflexivity)},\\
@_i j \leftrightarrow @_j i & \text{(symmetry)},\\
(@_i j \wedge @_j k) \rightarrow @_i k & \text{(transitivity), and}\\
@_i j \rightarrow (\varphi \leftrightarrow \varphi(i/j)) & \text{(substitution by identicals)}^2
\end{array}$$

for arbitrary nominals i, j, k and an arbitrary formula φ are tautologies of $\mathscr{H}(@,\downarrow,\mathsf{A})$. Nevertheless, the satisfiability problem of $\mathscr{H}(@)$ remains PSpace-complete [5]. $\mathscr{H}(\mathsf{A})$ is also decidable, its satisfiability problem is EXPTime-complete, while the innocent-looking $\mathscr{H}(\downarrow)$ –even though expressively weaker than first-order logic– is already undecidable.

2.1 Resolution in Modal and Hybrid Logics

A characteristic of modal logics is that they are usually computationally robust [51,31]. They have been extensively used in applications, specially in the area of verification via model checking [20]. But automated satisfiability testing in modal logics has also a long history and a large community, with a wide range of approaches. Not aiming to be comprehensive, we can mention calculi and implemented systems which are based on tableaux, translation into other languages (e.g., propositional, first-order and higher-order logics) and formalisms (e.g., automata), resolution, sequents, etc.

In particular, the field of resolution for modal logics was especially active during the 80s and beginning of the 90s. Fariñas del Cerro et al. presented some of the first results including resolution based calculi for a number of modal logics [24,25,23], a notion of Herbrand models [19], an extension of PROLOG with modal operators [26,15], and special resolution strategies for certain language fragments [11]. Also Mints [40,41] investigated resolution calculi for modal logics, and in particular their relation with Gentzen systems. These proposals usually describe *clausal* resolution methods, i.e., input formulas are first put into some kind of clausal form, before resolution takes place (a recent account of this tradition can be found in [42]). But *non-clausal* approaches (closer in spirit to the method we introduce in this article) were also investigated in, e.g., [1,2,27].

During the same period, a different approach to resolution for modal languages was developed by Ohlbach [44]. The idea in this case was to translate modal formulas into predicate logic preserving satisfiability, and then apply saturation-based (e.g., resolution, superposition, etc.) inference methods, taking advantage in this way, of the extensive work on first-order automated deduction [46]. Later, much work was devoted to define specialized translations [48,49,8,34,50] which would ensure better computational behavior and in some cases even termination. Recent publications had targeted very expressive modal languages,

[2] $\varphi(i/j)$ is the formula obtained by replacing *all* occurrences of i in φ by j. Notice that because j appears free in the antecedent, we know that it doesn't appear bound in φ, and hence the formula is indeed a tautology.

like the description logic *SHOIQ* [36]. The translation based approach has even been used as a meta-framework for developing various calculi for different modal logics, as presented in [3]. A survey on translation-based approaches can be found in [22]. Much attention has also been paid to developing saturation-based calculi for 'modal' fragments of first-order logics, like guarded fragments [28,21,35].

As we mentioned above, we are interested in investigating whether the general framework of saturation-based inference for first-order logic with equality can be *directly* adapted to modal logics, i.e., without first translating modal formulas into their first-order equivalent. In [7] a resolution based calculus for $\mathscr{H}(@,\downarrow,\mathsf{A})$ is proposed. The formulation of the calculus that we will present takes formulas in *negation normal form* (NNF), i.e., the negation operator can only be applied to atoms. As a consequence, $\vee$, $\langle\cdot\rangle$ and E become primitive symbols. We define the set of formulas $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$ as follows:

$$\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A}) ::= a \mid \neg a \mid \varphi \vee \varphi' \mid \varphi \wedge \varphi' \mid \langle r\rangle\varphi \mid [r]\varphi \mid @_i\varphi \mid \downarrow i.\varphi \mid \mathsf{A}\varphi \mid \mathsf{E}\varphi$$

where $a \in \mathsf{ATOM}$, $r \in \mathsf{REL}$, $i \in \mathsf{NOM}$ and $\varphi, \varphi' \in \mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$. A formula of the form $@_i\varphi$ will be called an *@-formula* and i will be called its *label*. For any formula in $\mathscr{H}(@,\downarrow,\mathsf{A})$, an equivalent formula in $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$ can be obtained in linear time. From now on, we will only consider formulas in NNF unless we indicate otherwise.

Like the resolution calculus for first-order logic, the hybrid resolution calculus works on sets of *clauses*. A clause, in this context, is a finite set of arbitrary $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$ @-formulas. A clause represents the disjunction of its formulas, but there's no additional restriction regarding the form of the formulas (i.e., they do not need to be literals). It is worth noting that to allow only @-formulas in a clause is not an expressivity limitation in terms of satisfiability: a formula φ is satisfiable if and only if for some arbitrary nominal i not occurring in φ, $@_i\varphi$ is satisfiable.

Given a formula $\varphi \in \mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$, we define $\mathit{ClSet}(\varphi) = \{\{@_i\varphi\}\}$, for i an arbitrary nominal not occurring in φ. $\mathit{ClSet}^*(\varphi)$, the saturated set of clauses for φ, is then defined as the smallest set that includes $\mathit{ClSet}(\varphi)$ and is closed under the rules of the resolution calculus $\mathbf{R}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ given in Figure 1, where $i, j \in \mathsf{NOM}$, $p \in \mathsf{PROP}$, $\varphi, \varphi_1, \varphi_2$ are arbitrary formulas of $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$ and C, D are arbitrary clauses[3]. In the antecedent of the PAR rule, $\varphi(i)$ indicates that the nominal i appears in φ.

We can group the rules in Figure 1 according to their role. The $\wedge$, $\vee$, @ and $\downarrow$ rules handle formula decomposition. The $\langle r\rangle$ and E rules both do a form of skolemization, assigning a new name (through a new nominal) to an element of the model which was existentially quantified. The RES rule is a ground version of the resolution rule for first-order logic, while the $[r]$ propagates information across modal contexts. The A rule is just unrestricted instantiation. Finally, the SYM, REF and PAR rules are the standard set of rules for equality handling in (function free) first-order logic resolution [13]. Actually, we will usually call formulas of the form $@_i j$ *equality statements* or, simply, equalities.

Rules $\langle r\rangle$ and E are clearly satisfaction-preserving (i.e., if a model satisfies the antecedent, then there is a model that satisfies the consequent); the rest of the rules preserve satisfaction even in the same model (i.e., if a model satisfies the antecedent, then the same model satisfies the consequent), in the case of the $\downarrow$ rule under the assumption that i does not occur bound in φ.

[3] For simplicity, the definition of $\mathit{ClSet}^*(\varphi)$ does not take into account the possibility of redundancy elimination as used in [14]. But it should be noted that the calculi we will present in Sections 4, 5 and 6 have the *reduction property for counterexamples* and the rules are *reductive* with respect to admissible orders (cf. [14]), and hence are compatible with both the trivial and the standard redundancy criteria.

$$\text{RES}\quad \frac{C\cup\{@_i\neg p\}\quad D\cup\{@_i p\}}{C\cup D} \qquad\qquad \text{REF}\quad \frac{C\cup\{@_i\neg i\}}{C}$$

$$\text{SYM}\quad \frac{C\cup\{@_j i\}}{C\cup\{@_i j\}} \qquad\qquad \text{PAR}\quad \frac{C\cup\{\varphi(i)\}\quad D\cup\{@_i j\}}{C\cup D\cup\{\varphi(i/j)\}}$$

$$\wedge\quad \frac{C\cup\{@_i(\varphi_1\wedge\varphi_2)\}}{C\cup\{@_i\varphi_1\}\ \ C\cup\{@_i\varphi_2\}} \qquad\qquad \vee\quad \frac{C\cup\{@_i(\varphi_1\vee\varphi_2)\}}{C\cup\{@_i\varphi_1,@_i\varphi_2\}}$$

$$[r]\quad \frac{C\cup\{@_i[r]\varphi\}\quad D\cup\{@_i\langle r\rangle j\}}{C\cup D\cup\{@_j\varphi\}} \qquad\qquad \langle r\rangle\quad \frac{C\cup\{@_i\langle r\rangle\varphi\}}{C\cup\{@_i\langle r\rangle j\}\ \ C\cup\{@_j\varphi\}}\ \dagger$$

$$\mathsf{A}\quad \frac{C\cup\{@_i\mathsf{A}\varphi\}}{C\cup\{@_j\varphi\}}\ \ddagger \qquad\qquad \mathsf{E}\quad \frac{C\cup\{@_i\mathsf{E}\varphi\}}{C\cup\{@_j\varphi\}}\ \dagger$$

$$@\quad \frac{C\cup\{@_i@_j\varphi\}}{C\cup\{@_j\varphi\}} \qquad\qquad \downarrow\quad \frac{C\cup\{@_i{\downarrow}j.\varphi\}}{C\cup\{@_i\varphi(j/i)\}}$$

Side conditions

$\dagger$ $j\in\mathsf{NOM}$ is *fresh*.
$\ddagger$ $j\in\mathsf{NOM}$ already occurs in the clause set.

Fig. 1 The Resolution Calculus $\mathbf{R}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$.

The construction of $\mathit{ClSet}^*(\varphi)$ is a correct and complete algorithm for satisfiability for $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$ (and hence for $\mathscr{H}(@,\downarrow,\mathsf{A})$ and all its sublanguages): φ is unsatisfiable if and only if the empty clause $\{\}$ is an element of $\mathit{ClSet}^*(\varphi)$ [7]. However, the size of $\mathit{ClSet}^*(\varphi)$ cannot be bounded in terms of the subformulas of φ because each application of the $\langle r\rangle$ and E rules introduces a new nominal. Thus, the construction of $\mathit{ClSet}^*(\varphi)$ is not necessarily a decision method for satisfiability. In Section 6 we will first show that indeed, we can obtain infinite derivations, even after imposing restrictions to some of the rules, and then discuss how to obtain a decidable resolution calculus for satisfiability in $\mathscr{H}(@)$.

A standard technique to regulate the generation of clauses in resolution for first-order logic is called *ordered resolution with selection functions* [14]. The general idea is to establish certain conditions under which it is safe to *choose* a literal from each clause such that rules are to be applied to a clause only to eliminate its chosen literal. The ordered resolution calculus with selection functions is refutationally complete for first-order logic when an order $\succ$ with certain properties is used (see [14]). In the following sections we develop similar strategies for $\mathbf{R}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$. Preliminary results have been presented in [9].

3 A unified framework for saturation-based theorem proving

A straightforward, naive implementation of a saturation-based calculus in which one exhaustively applies inferences to previously derived clauses will be hopelessly inefficient in all but

$$\textsf{Deletion}\quad \frac{N \cup M}{N}\ \dagger \qquad \textsf{Deduction}\quad \frac{N}{N \cup M}\ \ddagger$$

Side conditions

† $M \subseteq \mathscr{R}(N)$.
‡ $M \subseteq \mathsf{C}(\Gamma(N))$ where $\mathsf{C}(\Gamma(N))$ are the consequents of $\Gamma(N)$.

Fig. 2 Derivations for a calculus Γ with redundancy criterion $\mathscr{R}$.

the most trivial cases. In a clausal, saturation-based, refutational theorem-prover, each derived clause is a potential partial derivation of the empty clause and is increasing the search space. A partial derivation of the empty clause that is subsumed by another one is redundant and should be deleted to avoid useless computations. In most refutational provers, the deductive core accounts for a rather small part of the system, while most of its complexity derives from the implementation of redundancy elimination and simplification techniques.

While redundancy elimination techniques are crucial from a practical point of view, it is not a priori clear to what extent they can be performed without compromising refutational completeness. Bachmair and Ganzinger [14] address this issue by introducing a theoretical framework for saturation-based theorem proving, which we briefly present now.

Theorem proving is modeled in [14] as a series of *derivations* that transform a set of clauses by additions and removals (we only consider the case for *ground* clauses here; for general clauses, an additional lifting has to be done). Derivation rules are shown in Figure 2. Observe that this is a very abstract presentation, where the calculus is represented by a map Γ from a set of clauses to the set of all inferences that may be drawn from those clauses, and $\mathscr{R}$ is a *redundancy criterion* that maps a set of clauses N to a set of clauses *deemed redundant* by N (for simplicity, we have not listed some properties that a redundancy criterion must satisfy, e.g., removing redundant clauses preserves unsatisfiability, etc.).

A clause set is said to be *saturated up to redundancy* (with respect to Γ and $\mathscr{R}$) if all inferences in Γ with non-redundant premises are redundant in N, i.e., $\mathsf{C}(\Gamma(N \setminus \mathscr{R}(N))) \subseteq \mathscr{R}(N)$. It can be shown that saturation up to redundancy can be achieved by "fair" derivations, that is, derivations where no non-redundant inference is delayed indefinitely.

Bachmair and Ganzinger also define what they call the *standard redundancy criterion* and give general conditions under which a refutationally complete calculus Γ induces a system that is *derivationally complete*, i.e., every unsatisfiable set saturated up to redundancy contains the empty clause. In their words, the standard redundancy criterion "justifies most, if not all, of the common simplification and deletion techniques used in refutational theorem provers." Standard redundancy is defined for ordered calculi. Intuitively, a clause C is said to be *redundant with respect to a set N of clauses* if there are clauses $C_1, \ldots C_k$ in N such that $C_1, \ldots, C_k \models C$ and $C \succ C_i$ for all $1 \le i \le k$. Observe that the definition comprehends the most classical form of redundancy elimination: the *subsumption principle* [47].

Now, what are the general conditions that guarantee that the standard redundancy criterion turns a refutationally complete system into a derivationally complete one? What [14] shows is that any calculus that possess what they call the *reduction property for counterexamples* is refutationally complete and induces a derivationally complete system in conjunction with the standard redundancy criterion, provided the logic is compact and the underlying order $\succ$ is total and well-founded.[4] The upshot is that given a saturation-based refutational

[4] The reader will not find this exact formulation in [14], but it follows from the results presented there. Bachmair and Ganzinger first establish in §3 that ground ordered resolution for first-order logic has the reduc-

calculus, if one can establish its completeness by proving the reduction property for counterexamples, one gets an "adequacy for implementations" result for free. In a way, it can be regarded as a completeness proof scheme that comes with an added value.

In the following sections we will prove the refutational completeness of several saturation-based calculi for hybrid logics by showing they possess the reduction property for counterexamples. Completeness proofs in general tend to be long and technical, and ours are no exception, but being familiar with the proof-scheme used makes them easier to follow.

Let clauses be just collections (e.g., sets or multisets) of ground formulas, interpreted in a disjunctive way. A calculus will be presented as a collection of inference rules; when a rule has more than one premise, we assume that one of them is tagged as the main premise, the others will be called side premises. Finally, we take as given a satisfaction relation $\models$ defined between models and formulas, clauses, etc. A set of clauses N is called *saturated with respect to a set of rules R* if every clause obtained from N by the application of one of the rules in R is already in N. A set of clauses N is *inconsistent with respect to R* whenever the saturation of N with respect to R contains the empty clause, otherwise it is *consistent*. A saturation-based calculus R is *refutationally complete* (or *complete* for short) if every unsatisfiable set of clauses is inconsistent with respect to R.

A completeness proof can be reduced to showing that every saturated consistent clause set is satisfiable, i.e., has a model. One of the main ingredients of this proof strategy is a procedure that builds a *candidate model* from any consistent (but not necessarily saturated nor satisfiable) set of clauses N. Let I_N be the model obtained from a set N using such a procedure. What is ultimately shown is that if $I_N \not\models N$, then N is not saturated.

To prove this, the proof-scheme relies on a well-founded total order on clauses $\succ_c$. These two conditions guarantee that whenever N is consistent and $I_N \not\models N$, then there exists a minimum (with respect to $\succ_c$) clause $C \in N$ such that $I_N \not\models C$; we call C the *minimum counterexample of I_N*. The *reduction property for counterexamples* (with respect to $\succ$ and the candidate model building procedure) holds if for every N and every minimum counterexample C of I_N, there exists an inference from N with main premise C and a conclusion D such that $C \succ D$ and D is also a counterexample of I_N. Observe that this trivially implies that N is not saturated and, therefore, this property implies refutational completeness.

The proof of this property is typically split in two lemmas which trivially imply it:

Lemma (Main premise reduction lemma) *On every inference rule, every consequent is smaller than its main premise.*

Lemma (Counterexample lemma) *Let N be consistent, and let $C \in N$ be a clause such that $I_N \not\models C$; then there exists some valid inference with C as main premise and the side premises, if any, in N, such that for some consequent D, $I_N \not\models D$.*

We haven't said much about I_N yet. In a classical first-order setting, *Herbrand models* are used [32]. These are models whose domain is the set of (ground) syntactic terms of the language and that may be represented succinctly by a set of positive ground literals. Therefore, building a model from N in the classical setting amounts to generating a set of positive ground literals occurring in N.

tion property for counterexamples if $\succ$ is "admissible" (a standard notion in first-order logic theorem proving, which implies the mentioned conditions) and then show in §4 that if an inference system Γ has the reduction property for counterexamples for an admissible order $\succ$, then Γ is compatible with the standard redundancy criterion; but this proof only uses the fact that $\succ$ is total and well-founded (together with compactness of first-order logic). The last part of §4 is devoted to show how to deal with unification.

The model building procedure is relatively simple. Every clause $C \in N$ contributes *at most* one formula to I_N. Let ε_C be what C contributes to the model I_N, then ε_C is either a singleton set containing a (positive) literal, or the empty set. When ε_C is not empty, C is said to be a *productive clause*. Now, it is important to observe that the content of ε_C depends on the contributions of the clauses which are $\succ_c$-smaller than C. Its precise definition has to be carefully tailored in order to prove the *Counterexample lemma*. The set of contributions of clauses $\succ_c$-smaller then C we just mentioned (i.e. $\bigcup_{C \succ D} \varepsilon_D$) plays an important role throughout the proof and is usually called I_C.

The general structure of the completeness proof is essentially simple (and very elegant), but carrying it out usually relies on the following two rather technical lemmas:

Lemma (Downwards preservation lemma) *If $I_N \not\models C$, then $I_C \not\models C$.*

Lemma (Upwards preservation lemma) *Let D be the consequent of an inference whose main premise is C. If $I_N \not\models C$ and $I_C \not\models D$, then $I_N \not\models D$.*

Let's see the role these two lemmas usually play. Given a clause C such that $C \in N$ and $I_N \not\models C$, using the *Downwards preservation lemma* we can conclude that it is also the case that $I_C \not\models C$. But since $I_C = \bigcup_{C \succ D} \varepsilon_D$, one needs to show that some productive clause D must have contributed a formula to I_C that "made" C not true (of course, the details of this vary with every proof). This fact should make D a suitable side premise for an inference from C such that for at least one consequent E, $I_C \not\models E$. Finally, we need the *Upwards preservation lemma* to conclude $I_N \not\models E$.

For the calculi we will consider this is all what we need as we will always be working with ground clauses. In the case of calculi for first-order logic, the result for ground clauses must be extended to general clauses (containing variables). This is typically done by using *liftable orders*, i.e., orders which are invariant under substitution of variables by terms [14].

As a final remark, although we have presented the proof-scheme in terms of an order on clauses $\succ_c$, this is normally just an extension to clauses of a specially tailored, total, well-founded order on formulas $\succ_f$. This order on clauses should at least ensure that $C \succ_c D$ implies either $D = \{\}$ or $\max_{\succ_f}(C) \succeq_f \max_{\succ_f}(D)$ (where $\max_{\succ_f}(C)$ is the maximum formula in C according to $\succ_f$). Such order can be obtained, for example, using the *multiset extension order* [12].

4 Ordered hybrid resolution with selection functions

The calculus $\mathbf{R}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ shown in Figure 1, although sound and complete, is simply not suitable for a realistic theorem-prover implementation: because *every formula* in each clause may lead to some inference, the set of clauses will tend to grow in an unmanageable way for all but the most trivial cases. As was already mentioned at the end of Section 2.1, we want to establish conditions under which it is safe to restrict the attention to only one formula per clause.

A customary way of achieving this is using *ordered resolution*. An ordered resolution calculus uses an order on formulas to restrict which formulas in a clause may participate in inferences. Of course, certain admissibility conditions must be imposed on the order to ensure completeness.

For some applications (e.g., terminating heuristics for special fragments), though, one wishes to be able to impose specific conditions on the criteria used for the selection of formulas in clauses in a way that cannot be expressed using an order on formulas. Thus, some ordered resolution calculi provide a mechanism to optionally override the default order-based formula selection using so-called *selection functions*. Here too, some conditions have to be imposed on the selection functions in order to preserve completeness.

In the rest of this section, we turn $\mathbf{R}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ into a calculus of ordered resolution with selection functions and give an admissibility condition on orders with which we prove refutational completeness. Contrasting with the case of resolution in first-order logic where orders are partial to account for unification of non ground terms, the orders we require are total. Hence, at most one formula in each clause will be available for inferences[5].

We begin by formalizing our notion of selection function. Our focus is on selection functions that pick at most one formula per clause. In the case of first-order logic, selection formulas typically may choose only negative literals [14] and we will follow essentially the same approach. However, as we work with clauses which can contain arbitrary @-formulas from $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$, we will not use the concept of "negative literals" in defining selection functions but rather that of "not being a positive literal", where the set of positive literals PLIT is defined as

$$\mathsf{PLIT} ::= @_i j \mid @_i p \mid @_i \langle r \rangle j$$

for $i, j \in \mathsf{NOM}$, $p \in \mathsf{PROP}$ and $r \in \mathsf{REL}$. Observe that in first-order logic resolution, being a negative literal is the same as not being a positive literal.

Definition 3 (Selection function) A *selection function* S assigns to each clause C a set of @-formulas $S(C)$ such that $S(C) \subseteq C$, $|S(C)| \leq 1$ and $S(C) \cap \mathsf{PLIT} = \emptyset$.

Figure 3 defines the calculus $\mathbf{R}^{S\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$, which is parameterized over an order on formulas $\succ$ and a selection formula S. The main premise of the binary rules is the leftmost. We denote as $ClSet^{*}_{S\succ}(\varphi)$ the minimum set that contains $ClSet(\varphi)$ and is closed under the rules of $\mathbf{R}^{S\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$.

$\mathbf{R}^{S\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ differs from $\mathbf{R}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ only in its global and side conditions. The side conditions prevent certain redundant inferences by: i) enforcing a normal form on equalities (SYM and PAR rules); ii) making the choice of the main premise unique (PAR rule) and iii) avoiding useless skolemizations ($\langle r \rangle$ and E rules). The global conditions, on the other hand, ensure that only one formula in each clause may be involved in inferences. We will call this formula the *distinguished formula* of the clause.

Definition 4 (*max*$^{\succ}$ and *dist*$^{S\succ}$) Given an order $\succ$ and a selection function S, we define $max^{\succ}(C)$ as the maximum formula (with respect to $\succ$) in C, and $dist^{S\succ}(C)$ as the function such that $dist^{S\succ}(C) = \varphi$ whenever either $S(C) = \{\varphi\}$, or both $S(C) = \{\}$ and $max^{\succ}(C) = \varphi$.

In Section 4.1 we will define a class of orders for which we will guarantee refutational completeness in Section 4.3, using the notion of Herbrand model introduced in Section 4.2.

4.1 Admissible orders

In a strict sense, any $\succ$ such that $\mathbf{R}^{S\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ is refutationally complete would be an admissible order, but this notion of admissibility, undeniably general, would not be of much use. Instead, we will give the name "admissible" to a non-empty class of orders satisfying certain conditions that can be effectively checked; and we will later prove that any admissible order (in this sense) induces a complete calculus.

[5] This property could simplify implementations and result in efficiency gains.

RES	$\dfrac{C\cup\{@_i\neg p\}\quad D\cup\{@_i p\}}{C\cup D}$		REF	$\dfrac{C\cup\{@_i\neg i\}}{C}$	
SYM	$\dfrac{C\cup\{@_j i\}}{C\cup\{@_i j\}}$	†	PAR	$\dfrac{C\cup\{\varphi(i)\}\quad D\cup\{@_i j\}}{C\cup D\cup\{\varphi(i/j)\}}$	‡
$\wedge$	$\dfrac{C\cup\{@_i(\varphi_1\wedge\varphi_2)\}}{C\cup\{@_i\varphi_1\}\ C\cup\{@_i\varphi_2\}}$		$\vee$	$\dfrac{C\cup\{@_i(\varphi_1\vee\varphi_2)\}}{C\cup\{@_i\varphi_1, @_i\varphi_2\}}$	
$[r]$	$\dfrac{C\cup\{@_i[r]\varphi\}\quad D\cup\{@_i\langle r\rangle j\}}{C\cup D\cup\{@_j\varphi\}}$		$\langle r\rangle$	$\dfrac{C\cup\{@_i\langle r\rangle\varphi\}}{C\cup\{@_i\langle r\rangle j\}\ C\cup\{@_j\varphi\}}$	$\star$
A	$\dfrac{C\cup\{@_i\mathsf{A}\varphi\}}{C\cup\{@_j\varphi\}}$	$*$	E	$\dfrac{C\cup\{@_i\mathsf{E}\varphi\}}{C\cup\{@_j\varphi\}}$	$\star$
$@$	$\dfrac{C\cup\{@_i @_j\varphi\}}{C\cup\{@_j\varphi\}}$		$\downarrow$	$\dfrac{C\cup\{@_i{\downarrow}j.\varphi(j)\}}{C\cup\{@_i\varphi(j/i)\}}$	

Side conditions

† $i\succ j$
‡ $i\succ j$ and $\varphi(i)\succ @_i j$
$\star$ $\varphi\notin\mathsf{NOM}$ and $j\in\mathsf{NOM}$ is *fresh*.
$*$ $j\in\mathsf{NOM}$ already occurs in the clause set.

Global conditions

- in $C\cup\{\psi\}$, ψ is such that $S(C\cup\{\psi\})=\emptyset$ and $\{\psi\}\succ C$, or else $S(C\cup\{\psi\})=\{\psi\}$
- in $D\cup\{\psi\}$, ψ is such that $S(D\cup\{\psi\})=\emptyset$ and $\{\psi\}\succ D$.

Fig. 3 The Resolution Calculus $\mathbf{R}^{S\succ}[\mathcal{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$, for S a selection function and $\succ$ an order.

In order to make this paper self-contained, we include formal definitions for the following well-known orders (see, e.g., [12] for details).

Definition 5 A binary relation $\succ$ is called an *order* if it is transitive and irreflexive; if, additionally, for any two distinct elements x and y one of $x\succ y$ or $y\succ x$ holds, $\succ$ is said to be *total*. An order $\succ$ is called *well-founded* when there is no infinite chain $x_1\succ x_2\succ x_3\ldots$

Let $\succ$ be an order between formulas, and let's indicate with $\varphi[\psi]_p$ a formula φ where ψ appears at position p. We say that $\succ$ has the *subformula property* if $\varphi[\psi]_p\succ\psi$ whenever $\varphi[\psi]_p\neq\psi$, and that it is a *rewrite order* when $\varphi[\psi_1]_p\succ\varphi[\psi_2]_p$ iff $\psi_1\succ\psi_2$.

A well-founded rewrite order is called a *reduction order*, and if it also has the subformula property, it is called a *simplification order*.

We will typically work with a lifting to clauses $\succ_c$ of an order on formulas $\succ$. As was mentioned at the end of Section 3, we need $\succ_c$ to be total, well-founded and to satisfy that if $C\succ_c D$, then either $D=\emptyset$ or $max^{\succ}(C)\succeq max^{\succ}(D)$. We include, for the sake of completeness, a possible lifting.

Definition 6 For $\succ$ a total order on formulas, $\succ_c$ is the unique order on clauses such that $C \succ_c D$ if and only if $C \neq \emptyset$, and either

- $D = \emptyset$, or
- $max^{\succ}(C) \succ max^{\succ}(D)$, or
- $max^{\succ}(C) = max^{\succ}(D)$ and $C \setminus max^{\succ}(C) \succ_c D \setminus max^{\succ}(D)$.

Definition 6 is just a specialization of the multiset order to the case of finite sets. Therefore, $\succ_c$ is a total order, and well-founded whenever $\succ$ is well-founded [12]. From now on, we will use $\succ$ to denote both an order on formulas an its lifting to clauses.

Definition 7 (Admissible orders) We say an order $\succ$ over $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$-formulas is *admissible* (for $\mathbf{R}^{\text{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$) if it satisfies the following conditions, for all $\varphi, \psi \in \mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$ and all $i, j \in \mathsf{NOM}$:

A1) $\succ$ is a total simplification order
A2) $\varphi \succ i$ for all $\varphi \notin \mathsf{NOM}$
A3) if $\varphi \succ \psi$, then $@_i\varphi \succ @_j\psi$
A4) if ψ is a proper subformula of φ, then $\varphi \succ \psi(i/j)$
A5) $[r]i \succ \langle r\rangle j$.

Definition 7 is simply listing conditions that are used throughout the completeness proof. We shall motivate them by way of examples.

- Conditions A1 and A3 imply a notion of subformula property for @-formulas (e.g., $@_i\langle r\rangle\varphi \succ @_j\varphi$). This is used in the proof of the *Main premise reduction lemma*.
- Conditions A2 and A3 imply that equalities are the smallest @-formulas, and this will be important when proving the *Upwards and downwards preservation lemma*.
- Condition A4 is required to guarantee that $@_i{\downarrow}j.\varphi \succ @_i\varphi(j/i)$ in the proof of the *Main premise reduction lemma*. It is also used in the *Upwards preservation lemma*.
- Condition A5 is needed, for example, to guarantee that, in the $[r]$ rule, the side premise is smaller than the main premise and, thus, that the *Main premise reduction lemma* holds.

Lemma 1 (Main premise reduction for $\mathbf{R}^{\text{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$) *Let $\succ$ be an admissible order. If C is the main premise of an inference rule of $\mathbf{R}^{\text{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ and D is one of its conclusions, then $C \succ D$.*

Proof The proof is straightforward but rather tedious due to the numerous cases to consider. The Appendix contains all the details.

Finally, we only need to show the conditions in Definition 7 are not too restrictive, and that there actually exist orders satisfying them. We will exhibit one such order, based on the Knuth-Bendix order (KBO) [37]. In what follows, we will consider every $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$-formula as a ground term over the set of operators

$$O = \mathsf{PROP} \cup \mathsf{NOM} \cup \mathsf{REL} \cup \{\neg, \wedge, \vee, @, \downarrow, \langle\,\rangle, [\,], \mathsf{A}, \mathsf{E}\}$$

with the obvious arities, the only proviso being that the nominal argument of every @-formula is considered as the rightmost argument in the corresponding term (e.g., the formula $@_i\langle r\rangle p$ will correspond to the term $@(\langle\,\rangle(r,p),i)$).

Proposition 1 *Let $>$ be any total order over O and let $w : O \to \mathbb{N} \setminus \{0\}$ be any weight function such that*

1. $w(i) = w(j)$ *for all* $i, j \in \mathsf{NOM}$
2. $w(f) > w(i)$ *for all* $f \in O \setminus \mathsf{NOM}$*,* $i \in \mathsf{NOM}$
3. $w([\,]) > w(\langle\,\rangle)$*.*

Let $\succ_k$ *be the Knuth-Bendix order based on* $>$ *and* w*. Then,* $\succ_k$ *is an admissible order.*

Proof Since $>$ is total, $\succ_k$ is a total simplification order on ground terms [12]. Let $\tilde{w}(t)$ be the sum of the weights of all the symbols occurring in t. Condition 2 guarantees that $\tilde{w}(\varphi) > \tilde{w}(i)$ for all $\varphi \notin \mathsf{NOM}$ and, thus, A2 holds. For A3, observe that, from the definition of KBO, if $\varphi \succ_k \psi$ (and $\tilde{w}(\varphi) = \tilde{w}(\psi)$) then $@(\varphi, i) \succ @(\psi, j)$. If φ has a proper subformula ψ, it is because φ is of the form $f(\varphi_1, \ldots \varphi_k)$ and, from Condition 2 we have $w(f) > 0$ which implies $\tilde{w}(\varphi(\psi)) > \tilde{w}(\psi)$. But from Condition 1 we have $\tilde{w}(\psi) = \tilde{w}(\psi(i/j))$ and, thus, $\varphi \succ_k \psi(i/j)$, which establishes A4. Finally, A5 follows trivially from Conditions 1 and 3.

4.2 Herbrand models for $\mathscr{H}(@, \downarrow, \mathsf{A})$

We already said that, in order to carry out the completeness proof sketched in Section 3 we shall provide a suitable notion of Herbrand model for $\mathscr{H}(@, \downarrow, \mathsf{A})$. But before going into the details of this, we ought to give the abstract conditions we expect such models to satisfy.

There are two features of classical Herbrand models we want to mimic. First, we want Herbrand models to be syntactic in nature: in first-order logic, the domain of a Herbrand model is the set of all ground terms of the language (or a partition of that set if dealing with equality) and, thus, the interpretation function for constants and function symbols is trivial. Second, we want to mirror the fact that any set of ground first-order atoms Γ induces a Herbrand model H_Γ such that $H_\Gamma \models \Gamma$. With this in mind, we are now ready to define hybrid Herbrand models.

Definition 8 ($\sim_I$) Given $I \subseteq \mathsf{PLIT}$, define $\sim_I \subseteq \mathsf{NOM} \times \mathsf{NOM}$ as the reflexive, symmetric and transitive closure of $\{(i, j) \mid @_i j \in I\}$. $\mathsf{NOM}/_{\sim_I}$ is the set of equivalence classes of $\sim_I$, and $[i]_I$ is the equivalence class assigned to i by $\sim_I$. We will usually write $[i]$ instead of $[i]_I$ when I is clear from context.

Definition 9 (Hybrid Herbrand models) A *hybrid Herbrand model* is just a set $I \subseteq \mathsf{PLIT}$. Furthermore, let $\langle \mathsf{PROP}, \mathsf{NOM}, \mathsf{REL} \rangle$ be the signature of PLIT and $i \in \mathsf{NOM}$; we will say that $I, i \models \varphi$ iff $M^I, [i] \models \varphi$, where $M^I = \langle W^I, (r^I)_{r \in \mathsf{REL}}, V^I, g^I \rangle$ with

$$\begin{aligned} W^I &= \mathsf{NOM}/_{\sim_I} \\ r^I &= \{([i], [j]) \mid @_i \langle r \rangle j \in I\} \\ V^I(p) &= \{[i] \mid @_i p \in I\} \\ g^I(i) &= [i]. \end{aligned}$$

Summing up, we identify hybrid Herbrand models with sets of positive literals, and interpret them as hybrid models whose domain is a partition of the set of all nominals.

Proposition 2 *If* I *is a hybrid Herbrand model, then* $I \models I$*.*

Proof Straightforward from Definition 9.

4.3 Refutational completeness of $\mathbf{R}^{\mathsf{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$

We are now ready to prove that if $\succ$ is an admissible order, then $\mathbf{R}^{\mathsf{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ is refutationally complete. Thus, in what follows we take $\succ$ to be a fixed admissible order.

The only ingredient still missing from the sketch in Section 3 is the model-building procedure. Before going into its formal definitions, let us explain what we will be trying to achieve. Candidate models for N are hybrid Herbrand models defined using ε_C, i.e., the contribution of each clause C to the final model. Because we want every productive clause (i.e., clauses with a non-empty contribution) to be a potential side premise for a binary rule, we will stipulate that only clauses C such that $S(C) = \emptyset$ and $max^{\succ}(C) \in \mathsf{PLIT}$ may be productive. Moreover, in order to properly deal with equality, we require an additional technical property on every productive clause C: the contribution of C must not be reducible by paramodulation with another productive clause. A similar requirement is usually demanded in the proof of completeness for other paramodulation-based calculi (cf. [43]). Definition 12 properly formalizes this notion of *reducedness*, but observe that it must be necessarily defined along with ε_C in a mutually recursive way. In defining this reduced form, we will use a substitution of nominals by the smallest nominal in the equivalence class induced by a Herbrand interpretation which we now introduce.

Definition 10 **(σ_I)** Given a hybrid Herbrand interpretation I, we define the substitution of nominals by nominals:

$$\sigma_I = \{i \mapsto j \mid i \sim_I j \wedge (\forall k)((k \sim_I j \wedge k \neq j) \rightarrow k \succ j)\}.$$

In words, σ_I substitutes each nominal with the least nominal of its class, which is taken as the class representative. We now define the set SIMP of formulas that cannot be further simplified using unary rules (i.e., rules with only one premise).

Definition 11 (SIMP) The set of *simple formulas* of $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$ is defined as:

$$\mathsf{SIMP} ::= @_i j \text{ (with } i \succ j) \mid @_i p \mid @_i \neg a \mid @_i \langle r \rangle j \mid @_i [r] \varphi$$

where $i, j \in \mathsf{NOM}$, $p \in \mathsf{PROP}$, $a \in \mathsf{ATOM}$, $r \in \mathsf{REL}$ and $\varphi \in \mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})$.

We define next the candidate model building procedure. Observe that Definitions 12 and 13 are mutually recursive.

Definition 12 **(Reduced form)** Let C be a clause and $\varphi = max^{\succ}(C)$. If $\varphi \in \mathsf{SIMP}$ and either a) $\varphi \in \mathsf{PLIT}$ and $\varphi = \varphi\sigma_{I_C}$, or b) $\varphi = @_i[r]\psi$ and $i = i\sigma_{I_C}$, then we say that both φ and C are in *reduced form*.

Definition 13 **(I_N, I^C, I_C and ε_C)** Let N be an arbitrary set of clauses, C an arbitrary clause (not necessarily in N), and let $\varphi = max^{\succ}(C)$.

- I_N, a *candidate model* for N, is defined as $I_N = \bigcup_{C \in N} I^C$.
- I^C, the *partial interpretation of N above C* is defined as $I^C = I_C \cup \varepsilon_C$.
- I_C, the *partial interpretation of N below C* is defined as $I_C = \bigcup_{C \succ D} \varepsilon_D$.
- ε_C, *the contribution of C to the candidate model*, is defined as $\varepsilon_C = \{\varphi\}$ whenever it simultaneously holds that:
  1. $C \in N$
  2. C is in reduced form
  3. $\varphi \in \mathsf{PLIT}$

4. $I_C \not\models C$, and
5. $S(C) = \emptyset$.

and $\varepsilon_C = \emptyset$ otherwise. If $\varepsilon_C \neq \emptyset$ then we call C *productive*.

Note that, because of the admissibility conditions, equalities are the smallest @-formulas. Hence, if C is productive and $dist^{S\succ}(C)$ is an equality then every formula in C must also be an equality. Furthermore, if $dist^{S\succ}(C)$ is not an equality, then $\sigma_{I_C} = \sigma_{I_D}$ for all $D \succ C$.

From here on when not specified otherwise, N is taken to be an arbitrary but fixed set of clauses, and C an arbitrary but fixed clause not necessarily in N.

Lemma 2 (Downwards preservation for $\mathbf{R}^{S\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$) *If $I_N \not\models C$, then $I_C \not\models C$.*

Proof (sketch, full details in the Appendix) We prove the contrapositive form, so assume, for the sake of contradiction, that for some $\varphi \in C$, $I_C \models \varphi$ but $I_N \not\models \varphi$. Observe that it cannot be the case $\varphi \in \mathsf{PLIT}$. Now, consider the least $D \succeq C$ such that $I_D \models \varphi$ but $I^D \not\models \varphi$. From Definition 13 there are only three cases to consider: $\varepsilon_D = \{@_i j\}$, $\varepsilon_D = \{@_i p\}$ and $\varepsilon_D = \{@_i \langle r \rangle j\}$. We will only look here at the last case which implies that there exist ψ_1 and ψ_2 such that $[r]\psi_2$ is a subformula of ψ_1 and $\varphi = @_k \psi_1$. But, by conditions A1, A2 and A5 of Definition 7, $\psi_1 \succ [r]\psi_2 \succ \langle r \rangle j$, and, thus, we get $@_i \langle r \rangle j \succeq max^{\succ}(C) \succeq \varphi \succ @_i \langle r \rangle j$.

By requiring productive clauses to be in reduced form, we can give a syntactic description of equalities occurring in I_N that allows us to prove the *Upwards preservation lemma* (refer to the Appendix for the proof).

Lemma 3 *If $i\sigma_{I_N} \neq i$, then I_N contains only one equality where i occurs, and it is of the form $@_i j$ with $j = j\sigma_{I_N}$.*

Lemma 4 (Upwards preservation for $\mathbf{R}^{S\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$) *Let D be the consequent of an inference rule whose main premise is C. If $I_N \not\models C$ and $I_C \not\models D$, then $I_N \not\models D$.*

Proof (sketch, full details in the Appendix) Because of the *Main premise reduction lemma*, $max^{\succ}(C) \succeq \varphi$ for all $\varphi \in D$. Since $I_N \not\models C$, we already know $I_N \not\models max^{\succ}(C)$. Hence, we can reduce the proof to showing that, for any φ, if $max^{\succ}(C) \succ \varphi$ and $I_C \not\models \varphi$, then $I_N \not\models \varphi$. Now, suppose, for the sake of contradiction, that $E \succeq C$ is the least clause such that φ is true under I^E but false under I_E. Of the three alternatives, here we will only consider the most interesting one, namely, $\varepsilon_E = \{@_i j\}$.

Clearly, for this to be possible φ must be of the form $@_k l$. Now, by Lemma 3, we conclude that either $\varepsilon_E \subset \{@_k l, @_l k\}$, or $\varepsilon_E \subset \{@_k m, @_l m\} \subset I^E$. However, the latter cannot be true since that would imply $k \succ m$ and $l \succ m$ and, because $\succ$ is a rewrite order, we would have $@_k l \succ @_k m$ and $@_k l \succ @_l m$ (notice, for the second case, that if $k \succ l$ then $@_k l \succ @_k m \succ @_l m$, while, if $l \succ k$, then $@_k l \succ @_l k \succ @_l m$). Thus, $\varepsilon_E \subset \{@_k l, @_l k\}$ should hold. However, $\varepsilon_E = \{@_k l\}$ cannot be the case, since that would imply $@_k l \succeq max^{\succ}(C) \succ @_k l$. Finally, if $\varepsilon_E = \{@_l k\}$, then $l \succ k$ and $@_k l \succ @_l k \succeq \varphi \succ @_k l$.

An inspection of the above proof shows that we can actually assert a more general result: if $max^{\succ}(C) \succ \varphi$ and $I_C \not\models \varphi$ then $I_D \not\models \varphi$ for all $D \succeq C$. From this, we get the following:

Corollary 1 *If C is a productive clause and $\varphi \in C$ but $\varepsilon_C \neq \{\varphi\}$, then $I_D \not\models \varphi$ for all $D \succeq C$.*

Lemma 5 *Let $C \in N$ be such that $C \neq \{\}$ and $I_C \not\models C$. If C is not productive, then there exists an inference in $\mathbf{R}^{S\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ such that*

1. *C is the main premise*
2. *the side premise (if present) is productive, and*
3. *some consequent E is such that $I_C \not\models E$.*

Proof (sketch, full details in the Appendix) Let $\varphi = dist^{S\succ}(C)$. If $\varphi \notin \mathsf{SIMP}$, C is trivially the premise of some unary rule and the proposition holds. Now, suppose $\varphi \in \mathsf{SIMP}$ is not in reduced form; this means, using Lemma 3, that some clause D (with $C \succ D$) contributes an $@_i j$ for an i occurring in φ. It is easy to check that, in this case, PAR can be applied on D and C. Finally, if φ is in reduced form, it must be of the form $@_i\neg i$ (note that $@_i\neg j$ cannot be in reduced form if $I_C \models @_i j$ and $i \neq j$), $@_i\neg p$ or $@_i[r]\psi$. We show how to proceed in the last case using the $[r]$ rule; the remaining two cases are analogous.

For $I_C \not\models @_i[r]\psi$ to happen, it must be the case that, for some nominal j, $I_C, i \models \langle r\rangle j$ but $I_C, j \not\models \psi$. This implies, together with the fact that C is in reduced form, that $@_i\langle r\rangle k \in I_C$ for some k such that $I_C \models @_j k$. Therefore, there must exist a clause D such that $C \succ D$ and $\varepsilon_D = \{@_i\langle r\rangle k\}$ which, hence, may be the side premise in an instance of the $[r]$ rule with C as the main premise. Now, let $E = \{@_j\psi\} \cup C' \cup D'$, where $C' = C \setminus \{@_i[r]\psi\}$ and $D' = D \setminus \{@_i\langle r\rangle k\}$ be the consequent of the inference. $I_C \not\models E$ follows from:

1. $I_C \not\models C$ implies $I_C \not\models C'$,
2. $C \succ D$ implies (using Corollary 1) $I_C \not\models D'$, and
3. $I_C \models @_j k$ and $I_C \not\models @_k\psi$, implies $I_C \not\models @_j\psi$.

We can finally put all the pieces together as was planned in Section 3. Lemmas 2, 4 and 5 fit together nicely into a *Counterexample lemma* which, together with Lemma 1, gives us the completeness result.

Theorem 1 $\mathbf{R}^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ *has the reduction property for counterexamples and is, therefore, refutationally complete.*

Summing up, then, with Theorem 1 we have established refutational completeness of $\mathbf{R}^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ and, moreover, the proof was obtained by following the standard proof-scheme for first-order saturation-based methods. This ensures the calculus can be implemented using standard redundancy elimination techniques.

In the rest of this paper we will pursue the definition of a terminating direct resolution based calculus for $\mathscr{H}(@)$ and we will introduce two refinements of $\mathbf{R}^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$. The one in Section 5 will allow us to limit the paramodulation inferences needed to guarantee completeness. This time, however, the completeness proof will have to be less standard and more involved. Finally, this calculus will be extended in Section 6 with machinery to control the generation of witnesses by rules $\langle r\rangle$ and E. The completeness proof will be essentially the same, but additionally we will be able to prove termination for $\mathscr{H}(@)$.

5 Paramodulation restricted to labels: $\mathbf{R}_L^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$

From the proof of Lemma 5 we can see that refutational completeness is preserved even if paramodulation is restricted to SIMP formulas. What we will see now is that by adding a simple sound rule to the calculus and using a construction slightly more involved, one can repeat the above proof and establish the stronger result that paramodulation inferences can be further restricted to the following rule:

$$\frac{C \cup \{@_i\varphi\} \quad D \cup \{@_i j\}}{C \cup D \cup \{@_j\varphi\}}\ i \succ j, \varphi \succ j, @_i\varphi \in \mathsf{SIMP}.$$

That is, we need not consider any other nominal but the label i of the distinguished formula $@_i\varphi$ of the main premise and we don't have to replace other occurrences of i inside φ. This by itself is a nice property from a practical point of view. Moreover, by taking advantage of this restriction we will be able to define in Section 6 a terminating calculus for $\mathscr{H}(@)$.

In Figure 4 we define $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$. Observe that not only we replaced PAR by the aforementioned rule, but we added a new inference rule: SYM$^\neg$. The calculus would not be complete without this additional rule; this is witnessed by the following example.

Example 1 Consider the set $N = \{C_1,C_2\}$ where $C_1 = \{@_i j\}$ and $C_2 = \{@_j \neg i\}$ with an order such that $i \succ j$. N is evidently unsatisfiable so, if SYM$^\neg$ were not required for the completeness of $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$, one should be able to find a $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$-derivation of the empty clause that does not use the SYM$^\neg$ rule. However, we cannot use PAR$^@$ to replace i by j in C_2, and since $i \succ j$ we cannot use SYM on C_1 to derive $\{@_j i\}$ aiming to replace j by i in C_2. Without SYM$^\neg$ we would be stuck. But if we use it, we can derive $\{@_i \neg j\}$ from C_2, which can then be used along with C_1 to derive $\{@_j \neg j\}$ using the PAR$^@$ rule, and from there rule REF gives us the empty clause.

We now have to make changes to the completeness proof of Section 4 until everything fits together again. It must be said upfront that, in this case, the required notions and definitions are much less intuitive. We will try to motivate them by giving a short account of the problems we will have to face when adapting the proof.

Example 2 Let $i \succ j \succ k$ and let $N = \{C_1,C_2,C_3\}$ where $C_1 = \{@_j k\}$, $C_2 = \{@_i k, @_i j\}$, and $C_3 = \{@_i \neg j\}$. It follows that, for any admissible order, $C_3 \succ C_2 \succ C_1$; therefore, we expect C_1 to be productive and, thus, we should have $I_{C_2} = \{@_j k\}$.

If we use the definitions from the completeness proof for $\mathbf{R}^{\mathsf{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$, C_2 would have to be non-productive (since it is not in reduced form) and this would make it the minimum counterexample for I_N, only reducible by PAR with C_1. However, this would not be a valid inference in $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$. The notion of "reduced form" was introduced to characterize clauses that cannot be the main premise of a paramodulation inference using a productive clause as side premise. Hence, we can already see that we need to adjust the notion of "being in reduced form" in order to account for the fact that C_2 cannot be reduced by paramodulation. The natural way to do this is by demanding only the label of the maximum formula to be reduced (this notion will be called *weak reduced form* in Definition 15).

But here comes the tricky part. If C_2 becomes a reduced clause, it will also turn into a productive one. In that case, C_3 would be the minimum counterexample for I_N and the only inference we can draw from it is by using the PAR$^@$ rule on C_2 obtaining $D = \{@_i k, @_j \neg j\}$. However, now $I_N \models D$, and thus we don't obtain a new counterexample. A closer inspection of this example shows that it is actually the *Upwards preservation lemma* that is failing.

$$\text{SYM}^\neg \quad \frac{C \cup \{@_j \neg i\}}{C \cup \{@_i \neg j\}}\ \dagger \qquad\qquad \text{PAR}^@ \quad \frac{C \cup \{@_i \varphi\} \quad D \cup \{@_i j\}}{C \cup D \cup \{@_j \varphi\}}\ \ddagger$$

Side conditions

$\dagger$ $i \succ j$
$\ddagger$ $i \succ j$, $\varphi \succ j$ and $@_i\varphi \in \mathsf{SIMP}$

Fig. 4 $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ is obtained from $\mathbf{R}^{\mathsf{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ by replacing PAR by SYM$^\neg$ and PAR$^@$.

Summing up, in order to use the proof-scheme of Section 3 we need an *Upwards preservation lemma*. But from the above example, that implies that the following should hold:

$$I_N \models @_i j \quad I_N \models @_j k \quad I_N \not\models @_i k$$

which is simply not possible (because $\models (@_i j \wedge @_j k) \rightarrow @_j k$). To escape from this apparent dead end, what we do is to drop the $\models$ relation altogether and repeat the completeness proof of Section 4, but now in terms of a carefully tailored relation $|\!\approx$ for which we shall have $I_N \mathrel{|\!\approx} @_i j$ and $I_N \mathrel{|\!\approx} @_j k$ while $I_N \mathrel{\not|\!\approx} @_i k$.

Of course, we will also have to ensure that whenever $I_N \mathrel{|\!\approx} \varphi$, then $I_N \models \varphi$ too, thus from proving that every saturated consistent set of clauses is $|\!\approx$-satisfiable we shall infer that every saturated consistent set of clauses is also $\models$-satisfiable. Before moving to the definition of $|\!\approx$, it is worth observing that a similar example can be devised for relations.

Example 3 Let $i \succ j \succ k \succ l$ and let $N = \{C_1, C_2, C_3, C_4\}$ where $C_1 = \{@_i l\}$, $C_2 = \{@_j k\}$, $C_3 = \{@_l \langle r \rangle j, @_i \langle r \rangle k\}$, and $C_4 = \{@_l [r] \neg j\}$. This time any admissible order entails $C_4 \succ C_3 \succ C_2 \succ C_1$ and, clearly, C_1 and C_2 must be productive clauses. According to the notion of reducedness of Definition 12, C_3 would be reducible and, therefore non-productive and the minimum counterexample for I_N. However, no inference can be drawn from C_3. Just like in Example 2 everything suggests we need to consider C_3 as a reduced clause, since its distinguished formula is $@_l \langle r \rangle j$ and l is indeed reduced, but this also makes it productive and C_4 becomes the minimum counterexample for I_N. The only clause we can derive from C_4 is $D = \{@_j \neg j, @_i \langle r \rangle k\}$, but $I_N \models D$ (because $I_N \models @_i \langle r \rangle k$) and therefore we don't obtain a smaller counterexample as required in the completeness proof.

It turns out that Examples 2 and 3 are sufficiently general, in the sense that every other counterexample can be seen as an instance of one of these two. Essentially, we can say that the exhibited problem relates to some form of *aliasing* due to the presence of nominals: in a productive clause, there is some non-distinguished formula that also becomes true when the distinguished formula is included in the candidate model (e.g. $@_i k$ in C_2 of Example 2 and $@_i \langle r \rangle k$ in C_3 of Example 3); we shall call them *aliased formulas*.

Now, fortunately, given some candidate model I_N, we can give a syntactic characterization of all the potential *aliased formulas* even without knowledge of N. So, we will basically stipulate that $I \mathrel{|\!\approx} \varphi$ holds whenever $I \models \varphi$ and φ is not potentially aliased, according to I.

Definition 14 ($|\!\approx$) We define $|\!\approx$ as the largest relation between hybrid Herbrand models and @-formulas, such that:

1. $I \mathrel{|\!\approx} \varphi$ implies $I \models \varphi$
2. $I \mathrel{|\!\approx} @_i j$ iff $I \mathrel{|\!\approx} @_j i$
3. $I \mathrel{|\!\approx} @_i j$ and $i \succ j$ implies that for no k such that $I \models @_j k$ and $k \succ j$, $@_i k \in I$
4. $I \mathrel{|\!\approx} @_i \langle r \rangle j$ implies that for no k and l such that $I \models @_i k$, $I \models @_j l$ and $l \succ j$, $@_l \langle r \rangle k \in I$.

Revisiting Example 2 may help grasp the ideas behind Definition 14.

- $I_N \mathrel{|\!\approx} @_j k$ holds because $@_j k \in I_N$ and no other equality labeled with j occurs in I_N.
- $I_N \mathrel{|\!\approx} @_i j$ holds for analogous reasons.
- Now, observe that $I_N \mathrel{|\!\approx} @_i k$ does not hold because $I_N \models @_j k$, $@_i j \in I_N$ but $j \succ k$. The last part is crucial; it implies that $@_i j \succ @_i k$ and, hence, $@_i k$ is a potentially aliased formula in the clause that contributed $@_i j$ to I_N (in fact, in Example 2 it is aliased).

The slight asymmetry between cases 3 and 4 in Definition 14 is due to the fact that, as shown in Example 3, in the case of relations the labels of the maximum formula and the aliased formula may differ[6]. The remaining definitions are more natural.

Definition 15 (Weak reduced form) Let C be a clause and $\varphi = max^{\succ}(C)$. If $\varphi = @_i\psi$ is a formula of SIMP and $i = i\sigma_{I_C}$, then we say that both φ and C are in *weak reduced form*.

As in the previous section, let N be an arbitrary but fixed set of clauses.

Definition 16 (ε_C for $\mathbf{R}_L^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$) Let C be a clause (not necessarily in N) and let $\varphi = max^{\succ}(C)$. If it simultaneously holds that:

1. $C \in N$
2. C is in weak reduced form
3. $\varphi \in \mathsf{PLIT}$
4. $I_C \not\models\!\approx C$
5. $S(C) = \emptyset$

then $\varepsilon_C = \{\varphi\}$; otherwise, $\varepsilon_C = \emptyset$.

Lemma 6 (Main clause reduction for $\mathbf{R}_L^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$) *If C is the main premise of an inference rule of $\mathbf{R}_L^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ and D is one of its conclusions, then $C \succ D$.*

Proof For SYM$^{\neg}$, the property follows from its side-condition. For the rest of the rules, it follows from the Main clause reduction lemma for $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$.

Lemma 7 (Downwards preservation for $\mathbf{R}_L^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$) *If $I_N \not\models\!\approx C$, then $I_C \not\models\!\approx C$*

Proof The proof runs similar to that of the equivalent lemma for $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$. We take $D \succeq C$ to be the least clause such that $I_D \models\!\approx C$ but $I^D \not\models\!\approx C$. Now, it must be the case that for some $\varphi \in C$, $I_D \models\!\approx \varphi$, but this implies that $I_D \models \varphi$. At this point, one can copy almost verbatim the proof of Lemma 2 to conclude that it must also be the case that $I^D \models \varphi$. Hence, if $I^D \not\models\!\approx \varphi$, it must be because one of Conditions 3 or 4 of Definition 14 does not hold.

For the first case, assume $\varphi = @_i j$ and, thus, $I^D \not\models\!\approx @_i j$. Since we know $I_D \models @_i j$, it must be the case that for some $k \succ j$, $I^D \models @_j k$ and $@_i k \in I^D$. This opens up two possibilities:

1. $@_i k \in I_D$. Since $I_D \models @_i j$, $I_D \models @_j k$, but that would imply $I_D \not\models\!\approx @_i j$.
2. $\varepsilon_D = \{@_i k\}$. We know $I_D \models @_i j$, but since D is productive, it must be in weak reduced form, hence, $j\sigma_{I_D} = i$ which implies $j \succeq i$. But since $@_i k \in \mathsf{SIMP}$, $i \succ k$ and, by hypothesis, $k \succ j$, which leads to $j \succeq i \succ k \succ j$.

For the second case, let $\varphi = @_i\langle r\rangle j$. Since $I_D \models\!\approx @_i\langle r\rangle j$, I_D must contain formulas other than equalities and ε_D cannot be an equality. Hence, it must be the case that $\varepsilon_D = \{@_l\langle r\rangle k\}$ for l and k such that $l \succ j$, $I_D \models @_i k$ and $I_D \models @_j l$. But since D is in weak reduced form we have $l\sigma_{I_D} = l$ which implies $j\sigma_{I_D} = l$ and, thus, $j \succeq l$. This contradicts $l \succ j$.

Observe that as a corollary we get that if C is productive, then $I_N \models\!\approx C$, which given conditions 3 and 4 of Definition 14, was not obvious.

[6] At this point some readers may wonder if Definition 14 couldn't be made simpler, e.g., by reducing case 3 to "$I \models\!\approx @_i j$ implies $@_i j \in I$." We encourage those readers to verify that with the simpler definition no counterexample can be inferred from the minimum counterexample for I_N, when N is $C_1 = \{@_j k\}, C_2 = \{@_i k\}, C_3 = \{@_k j\}, C_4 = \{@_i j\}, C_5 = \{@_i \neg j\}$; with $i \succ j \succ k$.

Lemma 8 (Upwards preservation for $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$) *Let D be the consequent of an inference rule whose main premise is C. If $I_N \not\approx C$ and $I_C \not\approx D$, then $I_N \not\approx D$.*

Proof (sketch, full details in the Appendix) The proof follows the scheme of Lemma 4. Thus, let φ be such that $max^{\succ}(C) \succ \varphi$ and $I_C \not\approx \varphi$, and let $E \succeq C$ be the least clause such that $I_E \not\approx \varphi$ but $I^E \approx \varphi$. Of the three possible cases, $\varepsilon_E = \{@_i p\}$ is handled exactly like in the proof for Lemma 4. We sketch the procedure for the case where ε_E is an equality, the remaining case runs similarly.

If $\varepsilon_E = \{@_a b\}$ with $a, b \in \mathsf{NOM}$, then $a \succ b$, $a\sigma_{I_E} = a$ and φ has to be an equality. That is, for some $i, j \in \mathsf{NOM}$, $i \succ j$, then either $\varphi = @_i j$ or $\varphi = @_j i$. In any case, we have $\varphi \succeq @_i j$ and also $I \approx @_i j$ iff $I \approx \varphi$. We can arrive to a contradiction proving, by case analysis, that it cannot be the case $a \succ i$ nor $i \succ a$ nor $a = i$. The first two rely only on properties of admissible orders, so we will only cover the last case here.

Let us assume that ε_E is $\{@_i b\}$. If $j \succeq b$, then $@_i j \succeq @_i b = max^{\succ}(E) \succeq max^{\succ}(C) \succ \varphi \succeq @_i j$. Now suppose $b \succ j$. For this case, we will rely on Condition 3 of Definition 14. Since $I^E \approx @_i j$, it must be the case $I^E \models @_i j$. Now, from this and $@_i b \in I^E$, we get $I^E \models @_j b$, but since $b \succ j$, $@_i b \succ @_i j$ and, thus, we get the contradiction $I^E \not\approx @_i j$

From here we can essentially repeat all the steps that lead us to Theorem 1. The only additional step is to verify that if the distinguished formula of the minimum counterexample is of the form $@_i \neg j$ and is in weak reduced form, then we must have $j \succ i$ and, thus, the $\text{SYM}^{\neg}$ rule is applicable, which is straightforward (see the Appendix).

Theorem 2 $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ *has the reduction property for counterexamples and is, therefore, refutationally complete.*

6 A decision procedure for the satisfiability of $\mathscr{H}(@)$

We mentioned in Section 2 that $\mathscr{H}(@)$ is a decidable subsystem of $\mathscr{H}(@,\downarrow,\mathsf{A})$. In this section we show how the calculus $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ can be turned into a decision procedure for the satisfiability problem of $\mathscr{H}(@)$. We will introduce the necessary changes to ensure that, for any formula $\varphi \in \mathscr{H}^{\text{NNF}}(@)$, the set that extends $ClSet(\varphi)$ and is closed by the rules of the calculus is a finite set. If this condition holds, implementing an effective algorithm that computes this set in finite time is straightforward (e.g., using the "given clause algorithm" [52]) . Observe, however, that $\mathbf{R}_{L\varepsilon}^{\mathsf{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$, the calculus we will introduce next, will still contain rules that handle the full $\mathscr{H}(@,\downarrow,\mathsf{A})$ language; we will show it to be sound and complete for $\mathscr{H}(@,\downarrow,\mathsf{A})$ and terminating for $\mathscr{H}(@)$.

First of all, it should be noted that, even when restricted to formulas from $\mathscr{H}(@)$, the calculus $\mathbf{R}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ introduced in Figure 1 can trivially generate an infinite saturated set of clauses, as the $\langle r \rangle$ rule can be applied to formulas of the form $@_i\langle r \rangle j$ for $j \in \mathsf{NOM}$[7]. Although both $\mathbf{R}^{\mathsf{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ and $\mathbf{R}_L^{\mathsf{S}\succ}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ avoid this behavior, an infinite number of nominals can still be introduced by the interaction between the $[r]$, $\langle r \rangle$ and the paramodulation rules. As no other symbols but nominals are introduced during resolution, and given that formulas in consequents are never larger (in number of operators) than those in the antecedent, if we can control the generation of nominals we will ensure termination.

[7] Actually, just repetitive application of the $\langle r \rangle$ rule to the same clause leads to the generation of an infinite set. This can be easily avoided by applying the rule only once to each $\langle r \rangle$-formula in a clause.

There are essentially two ways in which an infinite number of nominals can be introduced by the rules of $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ (or $\mathbf{R}_L^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$) when applied to a $\mathscr{H}^{\mathrm{NNF}}(@)$-formula:

Type 1. A formula of the form $@_i\langle r\rangle\varphi$ introduces a new nominal which, in turn, contributes to the derivation of a new clause containing $@_i\langle r\rangle\varphi$. All of these new nominals are immediate successors of i and they are actually representing the same state in the model, but the calculus cannot detect it.

Type 2. There is a formula φ and an infinite sequence of distinct nominals $i_0, i_1, i_2, \ldots$ such that, for all $n \geq 0$, some $@_{i_n}\langle r\rangle\varphi_n$ in the saturated set introduces, by way of the $\langle r\rangle$ rule, the nominal i_{n+1}. The calculus is exploring a cycle in the model and cannot detect when to stop the search.

For concrete examples, try the rules of $\mathbf{R}_L^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ over the formulas $@_i\langle r\rangle p \wedge @_i[r](q \vee @_i\langle r\rangle p)$ and $@_i\langle r\rangle p \wedge @_i[r](i \wedge \langle r\rangle p)$ using any admissible order where i is the smallest nominal and $p \succ q$.

This suggests that, to ensure termination, we need to impose some control both on the nominals generated by the $\langle r\rangle$ rule and on the way chains of nominal successors are treated. We will do this by introducing a hybrid version of Hilbert's ε-operator and modifying the calculus to exploit it.

6.1 Hilbert's ε-operator and $\mathbf{R}_{L\varepsilon}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$

The ε-operator was introduced by Hilbert as part of his program to establish the consistency of arithmetic by finitary means. A first-order ε-term is of the form $\varepsilon x.\varphi(x)$ where $\varphi(x)$ is a formula and its intended meaning is: “some element e such that $\varphi(e)$ holds, or an arbitrary element if no such e exists” [33,38]. The ε-terms were later investigated in the context of linguistics, philosophy and non-classical logics. From the point of view of automated reasoning, ε-terms can be seen as an alternative to skolem functions [29], and will serve as witnesses for existentially quantified objects.

In first-order logic enriched with the ε-operator, the notion of *formula* and *term* become mutually recursive. In hybrid logics, on the other hand, the boundary between formula and term vanishes; hence, they are an interesting setting in which to introduce ε-operators. The only important difference between ε-terms and fresh nominals is that the former will keep a record of which formula generated the new witness.

We will enrich $\mathscr{H}(@,\downarrow,\mathsf{A})$ with two types of (hybrid) ε-terms. The simplest of them will be of the form $\varepsilon\varphi$, with the intended meaning “an element where φ holds or an arbitrary one, if no such element exists.” For every nominal or ε-term l, and every relation symbol r, we will also have terms of the form $\varepsilon\langle l,r,\varphi\rangle$ denoting “an r-successor of l where φ holds if such exists, or any element (not necessarily a successor of l) otherwise.” Observe that unlike their first-order cousins, hybrid ε-terms do not bind variables.

Definition 17 (Syntax of $\mathscr{H}(@,\downarrow,\mathsf{A},\varepsilon)$) Let $\mathscr{S} = \langle \mathsf{PROP}, \mathsf{NOM}, \mathsf{REL}\rangle$, we define

$$\begin{aligned}
\mathscr{H}(@,\downarrow,\mathsf{A},\varepsilon) &::= a \mid \neg\varphi \mid \varphi\wedge\varphi' \mid [r]\varphi \mid @_l\varphi \mid \downarrow i.\varphi \mid \mathsf{A}\varphi\\
\varepsilon\text{-}\mathsf{forms} &::= \varepsilon\varphi \mid \varepsilon\langle l,r,\varphi\rangle\\
\mathsf{LAB} &::= \mathsf{NOM}\cup\varepsilon\text{-}\mathsf{forms}\\
\mathsf{ATOM} &::= \mathsf{PROP}\cup\mathsf{LAB}
\end{aligned}$$

where $a \in \mathsf{ATOM}$; $i \in \mathsf{NOM}$; $l,m \in \mathsf{LAB}$; $r \in \mathsf{REL}$ and $\varphi,\varphi' \in \mathscr{H}(@,\downarrow,\mathsf{A},\varepsilon)$.

Observe that ε-terms can occur nested, as in $@_{\varepsilon\langle\varepsilon\langle i,r_1,\neg p\rangle,r_2,p\rangle)}(q \wedge [r_1]\varepsilon q)$. Elements of the set LAB of *labels* will be denoted $l,m,n,\ldots$ Derived connectives are defined as expected. Also, the subsets of $\mathscr{H}(@,\downarrow,\mathsf{A})$ that were used until now are lifted to $\mathscr{H}(@,\downarrow,\mathsf{A},\varepsilon)$ in a natural way, using elements of LAB instead of just nominals (e.g., $\mathsf{PLIT} ::= @_l m \mid @_l p \mid @_l\langle r\rangle m$, $\mathsf{SIMP} ::= @_l m \mid @_l p \mid @_l\langle r\rangle m \mid @_l[r]\varphi$, etc.).

Definition 18 (Semantics of $\mathscr{H}(@,\downarrow,\mathsf{A},\varepsilon)$) We shall call *pre-structure* to any tuple $\langle M,A\rangle$ where $M = \langle W,(r^M)_{r\in\mathsf{REL}},V,g\rangle$ is a conventional hybrid model, and $A : \varepsilon\text{-}\mathsf{forms} \to W$. The satisfaction relation $\models$ is defined as follows:

$$\begin{array}{ll}
\langle M,A\rangle,w \models p & \text{iff } w \in V(p),\ p \in \mathsf{PROP}\\
\langle M,A\rangle,w \models i & \text{iff } w = g(i),\ i \in \mathsf{NOM}\\
\langle M,A\rangle,w \models e & \text{iff } w = A(e),\ e \in \varepsilon\text{-}\mathsf{forms}\\
\langle M,A\rangle,w \models \neg\varphi & \text{iff } \langle M,A\rangle,w \not\models \varphi\\
\langle M,A\rangle,w \models \varphi_1 \wedge \varphi_2 & \text{iff } \langle M,A\rangle,w \models \varphi_1 \text{ and } \langle M,A\rangle,w \models \varphi_2\\
\langle M,A\rangle,w \models [r]\varphi & \text{iff } r^M(w,w') \text{ implies } \langle M,A\rangle,w' \models \varphi, \text{ for all } w' \in W\\
\langle M,A\rangle,w \models @_i\varphi & \text{iff } \langle M,A\rangle,g(i) \models \varphi, i \in \mathsf{NOM}\\
\langle M,A\rangle,w \models @_e\varphi & \text{iff } \langle M,A\rangle,A(e) \models \varphi, e \in \varepsilon\text{-}\mathsf{forms}\\
\langle M,A\rangle,w \models \downarrow i.\varphi & \text{iff } \langle M_i^w,A\rangle,w \models \varphi\\
\langle M,A\rangle,w \models \mathsf{A}\varphi & \text{iff } \langle M,A\rangle,w' \models \varphi, \text{ for every } w' \in W.
\end{array}$$

A pre-structure $\langle M,A\rangle$ will be called a model when the following two conditions hold: i) if $\langle M,A\rangle \models \mathsf{E}\varphi$ then $\langle M,A\rangle \models @_{\varepsilon\varphi}\varphi$; and ii) if $\langle M,A\rangle \models @_l\langle r\rangle\varphi$ then $\langle M,A\rangle \models @_l\langle r\rangle\varepsilon\langle l,r,\varphi\rangle$ and $\langle M,A\rangle \models @_{\varepsilon\langle l,r,\varphi\rangle}\varphi$. Whenever $\langle M,A\rangle \models @_l m$ implies $A(\varepsilon\langle l,r,\varphi\rangle) = A(\varepsilon\langle m,r,\varphi\rangle)$ we will say that $\langle M,A\rangle$ is *closed under renaming of labels.*

Proposition 3 *Let φ be a formula where no ε-term occurs. Then φ is satisfiable iff it is satisfiable by a model closed under renaming of labels.*

Proof The right-to-left implication is trivial. Now, for the other direction, since no ε-term occurs in φ, that means that $\langle M,A\rangle \models \varphi$ implies $M \models \varphi$. So we only need to check that given $M = \{W,(r^M)_{r\in\mathsf{REL}},V,g\}$ we can always pick an A' such that $\langle M,A'\rangle$ is closed under renaming of labels. For this, take any total and well-founded order over W, let $w_\perp$ be some fixed element of W and simply define:

$$A'(\varepsilon\varphi) = \begin{cases} \min\{w \mid M,w \models \varphi\} & \text{if } M \models \mathsf{E}\varphi\\ w_\perp & \text{otherwise}\end{cases}$$

$$A'(\varepsilon\langle l,r,\varphi\rangle) = \begin{cases} \min\{w \mid r^M(g(l),w) \text{ and } M,w \models \varphi\} & \text{if } l \in \mathsf{NOM} \text{ and } M \models @_l\langle r\rangle\varphi\\ \min\{w \mid r^M(A'(l),w) \text{ and } M,w \models \varphi\} & \text{if } l \text{ is an } \varepsilon\text{-term and } M \models @_l\langle r\rangle\varphi\\ w_\perp & \text{otherwise}\end{cases}$$

A' is well-defined and it is trivial to verify that $\langle M,A'\rangle$ is closed under renaming of labels.

As before, for the rest of this section we will work with formulas in NNF. In Figure 5 we introduce the rules of $\mathbf{R}^{\mathrm{S}\succ}_{L\varepsilon}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$, a calculus that we will prove complete for $\mathscr{H}(@,\downarrow,\mathsf{A})$ and terminating for the fragment $\mathscr{H}(@)$. This calculus differs from $\mathbf{R}^{\mathrm{S}\succ}_{L}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ in only a few aspects: i) the rules are written in terms of formulas which may be labeled by elements in LAB, ii) rules $\langle r\rangle$ and E now use ε-terms instead of fresh nominals and, iii) the $\mathrm{PAR}^{@}$ rule is replaced by three rules: $\mathrm{PAR}^{@\text{no-}\Diamond}$, $\mathrm{PAR}^{@\Diamond}_{\text{no-}\varepsilon}$ and $\mathrm{PAR}^{@\Diamond}_{\varepsilon}$. The first two are simply the $\mathrm{PAR}^{@}$ rule over a restricted domain. As we will later see, the last one

RES	$\dfrac{C \cup \{@_l \neg p\} \quad D \cup \{@_l p\}}{C \cup D}$		REF	$\dfrac{C \cup \{@_l \neg l\}}{C}$	
SYM	$\dfrac{C \cup \{@_m l\}}{C \cup \{@_l m\}}$	†	SYM$^{\neg}$	$\dfrac{C \cup \{@_m \neg l\}}{C \cup \{@_l \neg m\}}$	†
PAR$^{@\text{no-}\diamond}$	$\dfrac{C \cup \{@_l \varphi\} \quad D \cup \{@_l m\}}{C \cup D \cup \{@_m \varphi\}}$	‡			
PAR$^{@\diamond}_{\varepsilon}$	$\dfrac{C \cup \{@_l \langle r \rangle \varepsilon\langle l,r,\varphi\rangle\} \quad D \cup \{@_l m\}}{\begin{array}{c} C \cup D \cup \{\ @_m \langle r \rangle \varepsilon\langle m,r,\varphi\rangle\ \} \\ C \cup D \cup \{\ @_{\varepsilon\langle l,r,\varphi\rangle} \varepsilon\langle m,r,\varphi\rangle\ \} \end{array}}$	†	PAR$^{@\diamond}_{\text{no-}\varepsilon}$	$\dfrac{C \cup \{@_l \langle r \rangle n\} \quad D \cup \{@_l m\}}{C \cup D \cup @_m \langle r \rangle n}$	⋆
$\wedge$	$\dfrac{C \cup \{@_l (\varphi_1 \wedge \varphi_2)\}}{C \cup \{@_l \varphi_1\} \ C \cup \{@_l \varphi_2\}}$		$\vee$	$\dfrac{C \cup \{@_l (\varphi_1 \vee \varphi_2)\}}{C \cup \{@_l \varphi_1, @_l \varphi_2\}}$	
$[r]$	$\dfrac{C \cup \{@_l [r] \varphi\} \quad D \cup \{@_l \langle r \rangle m\}}{C \cup D \cup \{@_m \varphi\}}$		$\langle r \rangle_{\varepsilon}$	$\dfrac{C \cup \{@_l \langle r \rangle \varphi\}}{\begin{array}{c} C \cup \{\ @_l \langle r \rangle \varepsilon\langle l,r,\varphi\rangle\ \} \\ C \cup \{\ @_{\varepsilon\langle l,r,\varphi\rangle} \varphi\ \} \end{array}}$	$*$
A	$\dfrac{C \cup \{@_l \mathsf{A} \varphi\}}{C \cup \{@_m \varphi\}}$	♮	E_{ε}	$\dfrac{C \cup \{@_l \mathsf{E} \varphi\}}{C \cup \{@_{\varepsilon\varphi} \varphi\}}$	$*$
@	$\dfrac{C \cup \{@_l @_m \varphi\}}{C \cup \{@_m \varphi\}}$		$\downarrow$	$\dfrac{C \cup \{@_l {\downarrow} m.\varphi\}}{C \cup \{@_l \varphi(m/l)\}}$	

Side conditions

† $l \succ m$
‡ $l \succ m$, $\varphi \succ m$, $@_l \varphi \in \mathsf{SIMP}$ and $\varphi \neq \langle r \rangle n$
⋆ $l \succ m$ and $n \neq \varepsilon\langle l,r,\varphi\rangle$
$*$ $\varphi \notin \mathsf{LAB}$
♮ $j \in \mathsf{NOM}$ already occurs in the clause set.

Global conditions

- in $C \cup \{\psi\}$, ψ is such that $S(C \cup \{\psi\}) = \emptyset$ and $\{\psi\} \succ C$, or else $S(C \cup \{\psi\}) = \{\psi\}$
- in $D \cup \{\psi\}$, ψ is such that $S(D \cup \{\psi\}) = \emptyset$ and $\{\psi\} \succ D$

Fig. 5 The Resolution Calculus $\mathbf{R}^{S\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$, for S a selection function and $\succ$ an order.

is crucial in order to guarantee termination while PAR$^{@\diamond}_{\text{no-}\varepsilon}$ is not necessary when the input does not contain the $\downarrow$-operator.

We first prove soundness of the calculus, as the proof is slightly more involved than in the previous cases.

Theorem 3 $\mathbf{R}^{S\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ *is a sound calculus.*

Proof Let φ be an @-formula in $\mathscr{H}(@,\downarrow,\mathsf{A})$. There exists a model M such that $M \models \varphi$ iff, for any A, $\langle M,A\rangle \models \varphi$ and, from Proposition 3, this will happen iff there exists A' such that $\langle M,A'\rangle \models \varphi$ and $\langle M,A'\rangle$ is closed under variable renaming. But $\mathbf{R}^{S\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ is clearly sound with respect to the class of models closed under variable renaming, so the empty clause cannot be derived from a satisfiable formula.

We now move to refutational completeness. For this, we first need to extend the notion of admissible order to the new language.

Definition 19 (Admissible orders) We say an order $\succ$ over $\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A},\varepsilon)$-formulas is *admissible* for $\mathbf{R}^{\mathsf{S}\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ if it satisfies the following conditions, for all $i \in \mathsf{NOM}$, every $l,m \in \mathsf{LAB}$ and all $\varphi,\psi \in \mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A},\varepsilon)$:

A1) $\succ$ is a total simplification order
A2) $\varphi \succ l$ for all $\varphi \notin \mathsf{LAB}$
A3) if $\varphi \succ \psi$, then $@_l\varphi \succ @_m\psi$
A4) if ψ is a proper subformula of φ, then $\varphi \succ \psi(l/m)$
A5) $[r]l \succ \langle r\rangle m$
A6) $l \succ m$ implies $\varepsilon\langle l,r,\varphi\rangle \succ \varepsilon\langle m,r,\varphi\rangle$.

The reader should check that conditions A1 to A5 are essentially those of Definition 7 but generalized to LAB. It is straightforward to extend the Knuth-Bendix order we used in Proposition 1 to obtain an order that also satisfies Condition A6. Notice that this condition guarantees that the main premise of rule $\text{PAR}^{@\Diamond}_{\varepsilon}$ is greater than its consequents. With this, it is easy to see that the *Main premise reduction lemma* holds.

Theorem 4 $\mathbf{R}^{\mathsf{S}\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ *has the reduction property for counterexamples and is, therefore, refutationally complete.*

Proof The proof is an almost verbatim reproduction of the completeness proof we already did for $\mathbf{R}^{\mathsf{S}\succ}_{L}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$. The only thing to be adjusted is that now a set I of PLIT formulas shall denote the pre-structure $\langle I, A_I\rangle$, where $A_I(l) = [l]$, i.e., the equivalence class of l in I. Observe that when the least counterexample is not in weak-reduced form, the $\text{PAR}^{@\text{no-}\Diamond}$, $\text{PAR}^{@\Diamond}_{\varepsilon}$ and $\text{PAR}^{@\Diamond}_{\text{no-}\varepsilon}$ rules can guarantee that a new counterexample is derived.

Notice that the calculus $\mathbf{R}^{\mathsf{S}\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ is *not* complete for $\mathscr{H}(@,\downarrow,\mathsf{A},\varepsilon)$, but only for $\mathscr{H}(@,\downarrow,\mathsf{A})$. For example, the formula $@_ip \wedge @_{\varepsilon p}\neg p$ leads to a consistent clause set although it is unsatisfiable (since $@_{\varepsilon p}\neg p$ must be false, according to Definition 18, in models where there is a state making p true, which is exactly what $@_ip$ ensures). The catch in the proof of Theorem 4 is that the pre-structure $\langle I, A_I\rangle$ does not necessarily satisfy the conditions imposed in Definition 18 for it to be a model of $\mathscr{H}(@,\downarrow,\mathsf{A},\varepsilon)$.

6.2 $\mathbf{R}^{\mathsf{S}\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ is terminating for $\mathscr{H}(@)$

We now prove that there are admissible orders $\succ$ such that, when the input formula is in $\mathscr{H}^{\text{NNF}}(@)$, $\mathbf{R}^{\mathsf{S}\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ doesn't generate infinite saturated sets.

Definition 20 Define the function level : $\mathsf{LAB} \to \mathbb{N}$:

$$\begin{aligned}\text{level}(i) &= 0\\ \text{level}(\varepsilon\varphi) &= 0\\ \text{level}(\varepsilon\langle l,r,\varphi\rangle) &= \text{level}(l)+1.\end{aligned}$$

We say that $\succ$ *respects levels* if for every $l,m \in \mathsf{LAB}$, $\text{level}(l) > \text{level}(m)$ implies $l \succ m$.

We will prove that if an order $\succ$ is admissible for $\mathbf{R}^{\mathsf{S}\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ and respects levels then, for every $\varphi \in \mathscr{H}(@)$, the following conditions hold:

T1(φ): the set $\{l \mid \text{level}(l) = k \text{ and } l \text{ occurs in } \mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)\}$ is finite, for all $k \geq 0$.
T2(φ): the set $\{\text{level}(l) \mid l \text{ occurs in } \mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)\}$ is finite.

where $\mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)$ is the least set that contains $\mathit{ClSet}(\varphi)$ and is closed under the rules of $\mathbf{R}^{\mathsf{S}\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$. The reader can check that these conditions guarantee, respectively, that problems of Type 1 and 2 discussed above cannot occur. Condition T1 actually holds even for formulas of $\mathscr{H}(@,\mathsf{A})$. In what follows, $\succ$ is taken to be an arbitrary admissible order for $\mathbf{R}^{\mathsf{S}\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@,\downarrow,\mathsf{A})]$ that respects levels.

Theorem 5 *For every $\varphi \in \mathscr{H}^{\text{NNF}}(@,\mathsf{A})$, T1($\varphi$) holds.*

Proof First, three rather trivial observations:

1. If $\varepsilon\langle l,r,\psi\rangle$ occurs in $\mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)$, then so does l.
2. If $\varepsilon\langle l,r,\psi\rangle$ occurs in $\mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)$, then $\langle r\rangle\psi$ must be a subformula of φ.
3. If $\varepsilon\psi$ occurs in $\mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)$, then ψ must be a subformula of φ.

Observe that the last two do not hold if the $\downarrow$-operator occurs in φ[8]. Now, let us define $C^k = \{l \mid \text{level}(l) = k \text{ and } l \text{ occurs in } \mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)\}$, and proceed by induction on k. C^0 is finite: it contains the finitely many nominals in $\mathit{ClSet}(\varphi)$ plus, from Observation 3, finitely many terms of the form $\varepsilon\psi$. For the inductive case, let us suppose, that C^k is finite but C^{k+1} is not. By Observation 1, it must be the case that for some $l \in C^k$ there exist infinitely many $\langle r_0\rangle\psi_0, \langle r_1\rangle\psi_1, \langle r_2\rangle\psi_2\ldots$ such that $\varepsilon\langle l,r_i,\psi_i\rangle \in C^{k+1}$ for $i \geq 0$. However, this clearly contradicts Observation 2, for φ has only finitely many subformulas.

To prove that condition T2(φ) holds for $\varphi \in \mathscr{H}(@)$, we need to find an upper bound for the level of the labels that may appear in $\mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)$. The modal depth $d(\varphi)$ (as defined for the basic modal logic, i.e., the maximum nesting of diamonds and boxes, see [17]) gives us the desired bound. The proof relies heavily on the fact that, as long as the $\downarrow$-binder does not occur in the input formula, terms of the form $\varepsilon\langle l,r,\psi\rangle$ may occur only in restricted positions. The following lemma formalizes this statement (again, the property holds even for $\mathscr{H}^{\text{NNF}}(@,\mathsf{A})$).

Lemma 9 *For all $\varphi \in \mathscr{H}^{\text{NNF}}(@,\mathsf{A})$, $\varepsilon\langle l,r,\psi\rangle$ occurs in $\mathit{ClSet}^*_{\mathsf{S}\succ T}(\varphi)$ only in the following kind of formulas:*

1. *$@_m\varepsilon\langle l,r,\psi\rangle$, with $m \neq \varepsilon\langle l,r,\psi\rangle$*
2. *$@_l\langle r\rangle\varepsilon\langle l,r,\psi\rangle$*
3. *$@_{\varepsilon\langle l,r,\psi\rangle}\theta$, and if an ε-term occurs in θ, then $@_{\varepsilon\langle l,r,\psi\rangle}\theta$ is also a formula of kind 1 or 2.*

Proof The proof is by induction on the derivation of a formula where $\varepsilon\langle l,r,\psi\rangle$ occurs. For the base case, an ε-term simply cannot occur in φ. For the inductive case, consider the last rule used to derive a formula containing $\varepsilon\langle l,r,\psi\rangle$. We discuss only the few interesting cases. It cannot be the case that the SYM$^\neg$ rule generates $@_m\neg\varepsilon\langle l,r,\psi\rangle$ because the premise would have to be $@_{\varepsilon\langle l,r,\psi\rangle}\neg m$ and, since by inductive hypothesis m could not be an ε-term, $\text{level}(\varepsilon\langle l,r,\psi\rangle) > \text{level}(m)$ and this implies, by Definition 20, that $\varepsilon\langle l,r,\psi\rangle \succ m$ which contradicts the side-condition of the rule. For the PAR$^{@\text{no-}\Diamond}$ rule, the interesting case is when both premises are equalities but in that case the side condition guarantees that $@_m m$ cannot be derived. Finally, observe that, by inductive hypothesis, there are no suitable premises for the PAR$^{@\Diamond}_{\text{no-}\varepsilon}$ rule.

Incidentally, the above proof also shows that the PAR$^{@\Diamond}_{\text{no-}\varepsilon}$ rule only is required when the input formula contains the $\downarrow$-operator (since no suitable premises for this rule will be

[8] For examples of this, consider $\varphi = @_i\langle r\rangle{\downarrow}j.\langle r\rangle(j \wedge p)$ and $\varphi = @_i\langle r\rangle{\downarrow}j.\mathsf{E}\neg j$.

derived otherwise). The above lemma is roughly saying that, in $ClSet^*_{S \succ T}(\varphi)$, terms of the form $\varepsilon\langle l, r, \psi\rangle$ may occur only as labels of @-formulas, or as the right-hand-side of equalities and relations. However, in the last case they are restricted to this form: $@_l\langle r\rangle\varepsilon\langle l, r, \psi\rangle$ and we know that $\text{level}(\varepsilon\langle l, r, \psi\rangle) = \text{level}(l) + 1$. Hence, in order to show that T2(φ) holds, we only need to find a bound for the level of ε-terms occurring in labels and in the right-hand-side of equalities.

Theorem 6 *If $\varphi \in \mathscr{H}^{\text{NNF}}(@)$ then:*

- *$@_l\psi$ occurs in $ClSet^*_{S \succ T}(\varphi)$ implies* $\text{level}(l) + d(\psi) \leq d(\varphi)$
- *$@_l m$ occurs in $ClSet^*_{S \succ T}(\varphi)$ implies* $\text{level}(m) \leq d(\varphi)$.

The proof is carried out by (a rather long yet straightforward) induction on the derivation of formulas. It is presented in full detail in the Appendix. Observe that the theorem does not hold if there are occurrences of A, since after an application of the A rule it is not necessarily the case that $\text{level}(l) + d(\psi) \leq d(\varphi)$.

Corollary 2 *For all $\varphi \in \mathscr{H}(@)$, T2($\varphi$) holds.*

Since, for $\varphi \in \mathscr{H}(@)$, every formula in $ClSet^*_{S \succ T}(\varphi)$ is made of subformulas of φ and ε-terms, and from Theorem 5 and Corollary 2 there are only finitely many of the latter, it follows that $ClSet^*_{S \succ T}(\varphi)$ must be finite.

Theorem 7 $\mathbf{R}^{S\succ}_{L\varepsilon}[\mathscr{H}^{\text{NNF}}(@, \downarrow, \mathsf{A})]$ *is a decision procedure for the satisfiability of $\mathscr{H}(@)$.*

7 Conclusions

In this article we have investigated the labeled resolution calculus introduced in [7] for the hybrid logic $\mathscr{H}(@, \downarrow, \mathsf{A})$, paying special attention to the decidable fragment $\mathscr{H}(@)$. In particular, we described the changes needed to turn it into a calculus of ordered resolution with selection functions that is complete even under standard redundancy elimination (which is crucial for the development of efficient provers).

More generally, we aimed to show that the general framework of saturation-based reasoning for first-order logic with equality, as presented for example in [14,43], can be successfully adapted to calculi for other logics.

After introducing the notions of *selection functions* (Definition3) and *admissible orders* (Definition 7), we proved a general completeness result: the calculus $\mathbf{R}^{S\succ}[\mathscr{H}^{\text{NNF}}(@, \downarrow, \mathsf{A})]$ is complete for *any* admissible order and selection function (Theorem 1). Choosing different solving strategies is usually crucial to tune the behavior of a resolution prover to a particular application. The results presented in this paper show that, as long as the solving strategies can be expressed in terms of admissible orders and selection functions, they will be refutationally complete. We actually proved that the calculus possesses the reduction property for counterexamples, which guarantees that refutational completeness holds even under redundancy elimination and simplification techniques. In the terminology of Bachmair and Ganzinger, we established that the calculus is *derivationally complete.*

We then showed that completeness is preserved even when paramodulation is restricted to labels in formulas (Theorem 2). It is well-known that paramodulation needs to be tightly controlled to obtain realistic implementations, and Theorem 2 shows that the set of clauses one needs to consider as candidates for paramodulation can be greatly reduced. But more importantly, this result paves the way to a decision method for $\mathscr{H}(@)$. We show that, by restricting paramodulation rules and using suitable orders, we can obtain a calculus that decides $\mathscr{H}(@)$ (Theorems 3, 4 and 7).

Termination of saturation-based calculi is typically established showing that the number of clauses in any saturated clause set is finite. For example, ordered resolution can be shown to decide certain fragments of first-order logic defining orders and selection functions that ensure a bound in the size of terms that can appear in clauses. The first two labeled resolution calculi we considered do not have a notion analogous to "term structure" that we could use: on-the-fly skolemization introduced nominals, which possess no inner structure. This motivated the introduction of ε-terms. These are semantically close to nominals (as both are used to univocally identify elements) but they have internal structure. By restricting to orders that treat ε-terms in a special way (Definition 20) we bounded their size, and hence proved that saturated clause sets are always finite. Notice that the decidability –actually, the exact complexity– of $\mathscr{H}(@)$ is a known result.[9] What the current article provides is a direct (i.e., without translation into first-order logic) saturation-based decision method for $\mathscr{H}(@)$ with restrictions via selection functions and orders.

To close this article we will mention certain aspects of our usage of the framework of saturation-based reasoning that are worth discussing.

i) The calculi we introduced do not involve unification. Hence we are only concerned with the *ground* version of the saturation-based framework. *Liftable* orders (partial orders which are invariant under substitutions), which play an important role in the original, first-order formulation of the framework ensuring that the order of literals is preserved once unification takes place, are not relevant in our case.

ii) Also, because we are dealing with ground calculi and total orders, there is exactly one formula per clause that can participate in inferences and, therefore, there is no need for selection functions that select more than one literal.

iii) Since our clauses are disjunctions of arbitrary @-formulas, the typical restriction of selection functions selecting only "negative" literals is too strong in our case. Instead we require selection functions to pick a "non-positive literal." We see this as a generalization of the condition for the first-order case, where the notions of "negative" and "non-positive" literals coincide.

iv) A crucial component in the proof that establishes the reduction property for counterexamples is a method for building candidate models. To this aim, we developed a suitable notion of Herbrand model for hybrid logics. This notion depends intrinsically on hybrid logics being able to directly refer to elements in their domain, and express reachability and (dis)equality between elements.

v) In the case of first-order logic, the term "admissible order" has a well-defined meaning (cf. e.g., [14]). The importance of these orders lies in that they ensure refutational completeness of classical first-order resolution calculi. We have retained the name "admissible" for an order that ensures that the reduction property for counterexample holds. Our notion of "admissibility" for the labeled resolution calculi is more complex than

[9] The decidability of the satisfiability problem for $\mathscr{H}(@)$ was established in [5], where the problem is shown to be PSpace-complete. Decidability results for more expressive extensions of the language are also known: e.g., the complexity of the satisfiability problem for $\mathscr{H}(@)$ extended with the universal modality A, the difference modality D, and inverse modalities $\langle\cdot\rangle^{-}$ was shown to be ExpTime-complete in [6]. A 2ExpTime upper bound for the extended language actually follows from complexity results for guarded fragments of first-order logic with equality [30]. Ganzinger and de Nivelle give in [28] a superposition decision procedure for the guarded fragment with equality but without functions nor constants (apart from those introduced by skolemization).

the one for first-order logic resolution. This seems to be the price to pay for having calculi that works on arbitrary formulas (and hence, with a larger number of rules), instead of requiring formulas to be in some kind of simplified clausal form.

Given that hybrid logics are extensions of modal logics, our results apply also to a number of modal logics. Clearly, we can decide satisfiability of formulas in the basic modal language **K** [17] using the terminating resolution calculus for $\mathscr{H}(@)$. But in addition, many other standard modal logics (e.g., **T**, **4**, etc.) can be defined in $\mathscr{H}(@, \downarrow, \mathsf{A})$ (up to satisfiability preservation), and hence our results also produce complete (possible non-terminating) calculi for them. More interestingly, modal logics can be defined in a modular way (i.e., new logics can be defined by the additions of new modal operators). These new modal operators can be handled by introducing new rules to the calculi we presented (as typically done, for example, in tableaux calculi for modal logics). If these new rules also satisfy that consequents are smaller than premises, and are enough to handle every counterexample to a candidate model then derivational completeness will also hold for the new calculus.

Appendix: Detailed proofs

Lemma 1 (Main premise reduction for $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$) *Let $\succ$ be an admissible order. If C is the main premise of an inference rule of $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ and D is one of its conclusions, then $C \succ D$.*

Proof Lemma 1 is a consequence of lemmas A1, A2 and A3 below. By Lemma A3, $dist^{S\succ}(C) \notin D$ and from Lemma A1 and Lemma A2 it follows that if $\varphi \in D$ and $\varphi \succ dist^{S\succ}(C)$, then $\varphi \in C$, thus, $C \succ D$.

Lemma A1 *Let C be the main premise of an inference rule of $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ and D one of its conclusions. If $\varphi \in D$ is a formula that does not occur in any of the premises, then $dist^{S\succ}(C) \succ \varphi$.*

Proof Let us consider each rule:

$\wedge, \vee$ Since $\succ$ is a simplification order, $(\varphi_1 \wedge \varphi_2) \succ \varphi_n$ where $n \in \{1,2\}$ and, thus, $@_i(\varphi_1 \wedge \varphi_2) \succ @_i\varphi_n$. The case for $\vee$ is analogous.

$[r]$ Because of the subformula property, $[r]\varphi \succ \varphi$, thus, by Condition A3 of Definition 7, $@_i[r]\varphi \succ @_j\varphi$.

$\langle r\rangle$ A variation of the former proof establishes $@_i\langle r\rangle\varphi \succ @_j\varphi$. The remaining case, $@_i\langle r\rangle\varphi \succ @_i\langle r\rangle j$, follows from Condition A2 and the side condition of the inference rule, that guarantees $\varphi \notin \mathsf{NOM}$.

A, E Analogous to the $\wedge$ rule.

$@$ By the subformula property, $@_i@_j\varphi \succ @_j\varphi$.

$\downarrow$ By Condition A4, we have $\downarrow j.\varphi \succ \varphi(j/i)$ which, entails $@_i\downarrow j.\varphi \succ @_i\varphi(j/i)$.

SYM Condition A3 of Definition 7 guarantees $@_ji \succ @_ij$ whenever $i \succ j$.

PAR If $j \succ i$, then it must be the case $\varphi(j) \succ \varphi(j/i)$ since $\succ$ is a rewrite order.

Lemma A2 *Let clauses C and D be, respectively, the main and side premises of a binary inference rule of $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$. Then $dist^{S\succ}(C) \succ max^{\succ}(D)$ and, therefore, $\{dist^{S\succ}(C)\} \succ D$.*

Proof In the case of the PAR rule this is true by the side condition. If the inference is an instance of the RES rule, then, since the admissible order has the subformula property,$\neg p \succ p$ for all $p \in \mathsf{PROP}$ and thus, because $\succ$ is a rewrite order, $@_i\neg p \succ @_ip$.

The interesting case is the $[r]$ rule, where we must verify that $@_i[r]\psi \succ @_i\langle r\rangle j$. By Condition A2 of Definition 7, there exists a nominal k such that $\varphi \succeq k$ and, thus, $[r]\varphi \succeq [r]k$. Finally, by Condition A5, $[r]\varphi \succeq [r]k \succ \langle r\rangle j$.

Lemma A3 *If C is the main premise of an inference rule of $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ one of whose conclusions is D, then $dist^{S\succ}(C) \notin D$, hence $C \not\subseteq D$.*

Proof On unary inferences this is self-evident. Now, on a binary inference with side premise C' we know from Lemma A2 that $\{dist^{S\succ}(C)\} \succ C'$, so $dist^{S\succ}(C) \notin C'$. From this and Lemma A1 it follows that $dist^{S\succ}(C) \notin D$.

Lemma 2 (Downwards preservation for $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$) *If $I_N \not\models C$, then $I_C \not\models C$.*

Proof We will prove the contrapositive: if $I_C \models C$, then $I_N \models C$. For the simpler case, if C contains a formula in PLIT that is true under I_C, then, since $I_C \subseteq I_N$, C must be true under I_N.

Now suppose for the sake of contradiction, that there is in C some formula $\varphi \notin \mathsf{PLIT}$ that is true under I_C but not under I_N. Because of compactness, there must exist some $D \succeq C$ such that $I_D \models \varphi$ but $I^D \not\models \varphi$; furthermore, since $\succ$ is total and well-founded, we can ask D to be the minimum such clause. This means that D contributes a formula that modifies I_D in a way that renders φ false under I^D. From Definition 13 we know that there are only three possibilities: $\varepsilon_D = \{@_i j\}$, $\varepsilon_D = \{@_i p\}$ or $\varepsilon_D = \{@_i \langle r \rangle j\}$.

Case $\varepsilon_D = \{@_i j\}$. If $\varphi \notin \mathsf{PLIT}$, then, because of conditions A2 and A3 of Definition 7, $max^{\succ}(C) \succeq \varphi \succ @_i j$, so it cannot be the case $D \succ C$.

Case $\varepsilon_D = \{@_i p\}$. It is clear that in this case, φ should be of form $@_j \psi$ and p would have to occur with a negative polarity in ψ. But, that means that p would have to be a proper subformula of ψ, so, $\varphi \succ p$ and, thus, $@_i p \succeq max^{\succ}(C) \succeq @_j \psi \succ @_i p$.

Case $\varepsilon_D = \{@_i \langle r \rangle j\}$. In this case, there should exist formulas ψ_1 and ψ_2 such that $[r]\psi_2$ is a subformula of ψ_1 and $\varphi = @_k \psi_1$. But, by conditions A1, A2 and A5, $\psi_1 \succ [r]\psi_2 \succ \langle r \rangle j$ (note that if $\psi_2 \notin \mathsf{NOM}$, $[r]\psi_2 \succ [r]j \succ \langle r \rangle j$), so $@_i \langle r \rangle j \succeq max^{\succ}(C) \succeq \varphi \succ @_i \langle r \rangle j$.

Lemma 3 *If $i\sigma_{I_N} \neq i$, then I_N contains only one equality where i occurs, and it is of the form $@_i j$ with $j = j\sigma_{I_N}$.*

Proof We will first show that I_N cannot contain two distinct equalities of the form $@_i k$ and $@_i l$. Next, we prove that if $@_i j$ occurs in I_N, then $j = j\sigma_{I_N}$. These two facts together further guarantee that two equalities of the form $@_k i$ and $@_i l$ cannot occur simultaneously in I_N.

Suppose, for the sake of contradiction, that two distinct equalities $@_i k$ and $@_i l$ occur in I_N, and let C and D be productive clauses contributing each equality respectively, with $C \succ D$. Under these assumptions, it follows that $\varepsilon_C = \{@_i k\}$ and $@_i l \in I_C$. From Definition 13, $@_i k$ must be in reduced form so, from Definition 12, it must be the case that $i = i\sigma_{I_C}$. However, if $@_i l \in I_C$, this cannot be true, for $i \succ l$.

We now turn to the second part of the proof, and we reason once again by contradiction. Suppose $@_i j \in I_N$ and yet $j\sigma_{I_N} \neq j$. If this is the case, there must be two clauses C and D such that C produces $@_i j$ while D contributes $@_j k$, thus, $i \succ j \succ k$. From Condition A3 of Definition 7 we infer that $@_i j \succ @_j k$ so $C \succ D$, hence $@_j k \in I_C$ and, once again, C cannot be a productive clause since $@_i j$ would not be in reduced form.

Lemma 4 (Upwards preservation for $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$) *Let D be the consequent of an inference rule whose main premise is C. If $I_N \not\models C$ and $I_C \not\models D$, then $I_N \not\models D$.*

Proof Because of the *Main premise reduction lemma*, $max^{\succ}(C) \succeq \varphi$ for all $\varphi \in D$. Since $I_N \not\models C$, we already know $I_N \not\models max^{\succ}(C)$. Hence, we can reduce the proof to showing that, for any φ, if $max^{\succ}(C) \succ \varphi$ and $I_C \not\models \varphi$, then $I_N \not\models \varphi$. Now, suppose, for the sake of contradiction, that $E \succeq C$ is the least clause such that φ is true under I^E but false under I_E. By Definition 13, there are only three cases we have to consider:

Case $\varepsilon_E = \{@_i j\}$. Clearly, this is only possible if φ is an equality too; thus, let $\varphi = @_k l$. Now, by Lemma 3, we conclude that either $\varepsilon_E \subset \{@_k l, @_l k\}$, or $\varepsilon_E \subset \{@_k m, @_l m\} \subset I^E$. However, the latter cannot be true since that would imply $k \succ m$ and $l \succ m$ and, because $\succ$ is a rewrite order, we would have $@_k l \succ @_k m$ and $@_k l \succ @_l m$ (notice, for the second case, that if $k \succ l$ then $@_k l \succ @_k m \succ @_l m$, while, if $l \succ k$, then $@_k l \succ @_l k \succ @_l m$). Thus, $\varepsilon_E \subset \{@_k l, @_l k\}$ should hold. However, $\varepsilon_E = \{@_k l\}$ cannot be the case, since that would imply $@_k l \succeq max^{\succ}(C) \succ @_k l$. Finally, if $\varepsilon_E = \{@_l k\}$, then $l \succ k$ and $@_k l \succ @_l k \succeq \varphi \succ @_k l$.

Case $\varepsilon_E = \{@_i p\}$. If this were the case, p would have to be a subformula of φ. It cannot be the case that $\varphi = @_k p$, for that would mean that $k\sigma_{I_E} = i$, which implies $k \succeq i$ and $\varphi \succeq @_i p \succeq max^{\succ}(C) \succ \varphi$. If, on the other hand, $\varphi = @_k \psi$ with $\psi \succ p$, then $\varphi \succ @_i p \succeq max^{\succ}(C) \succ \varphi$.

Case $\varepsilon_E = \{@_i \langle r \rangle j\}$. In this case, $\langle r \rangle \psi$ would have to be a subformula of φ for some formula ψ. If $\psi \notin \mathsf{NOM}$, then, since $\langle r \rangle \psi \succ \langle r \rangle j$, $\varphi \succ @_i \langle r \rangle j \succeq max^{\succ}(C) \succ \varphi$. Now suppose that $\psi = k$ for some nominal k. If φ is of the form $@_l \langle r \rangle k$ and is true under I^E because of $@_i \langle r \rangle j$, then it must be the case $l\sigma_{I_E} = i$ and $k\sigma_{I_E} = j$. However, that would imply $l \succeq i$, $k \succeq j$ and, thus, $\varphi \succeq @_i \langle r \rangle j \succeq max^{\succ}(C) \succ \varphi$. Finally, suppose $\varphi = @_l \psi'$, where $\langle r \rangle k$ is a proper formula of ψ'. Now, by Condition A4 of Definition 7, $\psi' \succ \langle r \rangle j$ and, thus, $\varphi \succ @_i \langle r \rangle j \succeq max^{\succ}(C) \succ \varphi$.

Lemma 5 *Let $C \in N$ be such that $C \neq \{\}$ and $I_C \not\models C$. If C is not productive, then there exists an inference in $\mathbf{R}^{\mathrm{S}\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ such that*

1. *C is the main premise*
2. *the side premise (if present) is productive, and*
3. *some consequent E is such that $I_C \not\models E$.*

Proof We only need to prove that for each C satisfying the hypothesis, there always exists an inference on C that generates a suitable consequent. Let $\varphi = dist^{S\succ}(C)$, the following case analysis shows that the lemmas A4, A5, A6 and A7 below handle all possible cases:

- The case where $\varphi \notin \mathsf{SIMP}$, is handled by Lemma A4.
- Otherwise, if $\varphi \in \mathsf{SIMP}$,
  - Lemma A5 handles the case where φ is not in reduced form.
  - If, on the other hand, φ is in reduced form,
    - φ cannot be in PLIT, for in that case $S(C) = \{\}$ and C would have to be a productive clause;
    - the case where φ is of the form $@_i\neg i$ is trivial by using the REF rule;
    - φ cannot be of the form $@_i\neg j$ with j a nominal other than i because $@_i\neg j$ false under I_C would imply, using Lemma 3, $\{@_ik, @_jk\} \subseteq I_C$ for some nominal k and, thus, φ would not be in reduced form;
    - Lemma A7 handles the case where φ is of the form $@_i\neg p$;
    - Lemma A6 handles the case where φ is of the form $@_i[r]\psi$.

Lemma A4 *If $C \neq \{\}$, $I_C \not\models C$ and $dist^{S\succ}(C) \notin \mathsf{SIMP}$, then C may be the premise of an unary inference of $\mathbf{R}^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ one of whose conclusions is false under I_C too.*

Proof We only consider the $\langle r\rangle$ rule (the rest are either trivial or analogous). Suppose C is false under I_C with $dist^{S\succ}(C) = @_i\langle r\rangle\psi$ and $\psi \notin \mathsf{NOM}$. The consequents of the $\langle r\rangle$ rule would be in this case $D = C' \cup \{@_i\langle r\rangle j\}$ and $E = C' \cup \{@_j\psi\}$ for some nominal j, where $C' = C \setminus dist^{S\succ}(C)$. Since C' is false under I_C, we only need to verify that either $@_i\langle r\rangle j$ or $@_j\psi$ is a false under I_C. However, this is evident, since otherwise, $@_i\langle r\rangle\psi$ would have to be true under I_C, and that would contradict our first assumption.

Lemma A5 *If $C \neq \{\}$, $I_C \not\models C$ and $dist^{S\succ}(C) \in \mathsf{SIMP}$ is not in reduced form, then there exists a productive clause D such that there is an instance of the PAR rule with C and D as premises whose conclusion is false under I_C too.*

Proof Let $\varphi = dist^{S\succ}(C)$. Since $\varphi \in \mathsf{SIMP}$ but is not in reduced form, from Definition 12, it follows that $\varphi \neq \varphi\sigma_{I_C}$. That is, for some nominal i occurring in φ, $i \neq i\sigma_{I_C}$. Therefore, there exists some clause $D \succ C$ such that $\varepsilon_D = \{@_ij\}$. Now, if φ is not an equality, then trivially $\varphi \succ @_ij$; otherwise, since it would not be possible $S(C) = \{\varphi\}$ (for equalities are in PLIT), we conclude $max^{\succ}(C) = \varphi$ and, thus, $\varphi \succ @_ij$. This means that C and D can be the premises of the PAR rule. Now, let $E = \{\varphi(i/j)\} \cup C' \cup D'$, where $C' = C \setminus \{\varphi\}$ and $D' = D \setminus \{@_ij\}$ be the consequent of the inference. $I_C \not\models E$ follows from:

1. $I_C \not\models C$ implies $I_C \not\models C'$,
2. $C \succ D$ implies $I_C \not\models D'$ (using Corollary 1), and
3. $I_C \models @_ij$ and $I_C \not\models \varphi$ implies $I_C \not\models \varphi(i/j)$.

Lemma A6 *If $C \neq \{\}$, $I_C \not\models C$ and $dist^{S\succ}(C) = @_i[r]\varphi$ is in reduced form, then there exists a productive clause D such that C and D may be the premises of an instance of the $[r]$ rule whose consequent is false under I_C too.*

Proof In this case, $I_C \not\models C$ is only possible if for some nominal j it simultaneously holds $I_C, i \models \langle r\rangle j$ and $I_C, j \not\models \varphi$. This implies, together with the fact that C is in reduced form, that $@_i\langle r\rangle k \in I_C$ for some k such that $I_C \models @_jk$. For this to happen, there must exist a clause $C \succ D$ and $\varepsilon_D = \{@_i\langle r\rangle k\}$ which, hence, may be the side premise in an instance of the $[r]$ rule with C as the main premise. Now, let $E = \{@_j\varphi\} \cup C' \cup D'$, where $C' = C \setminus \{@_i[r]\varphi\}$ and $D' = D \setminus \{@_i\langle r\rangle k\}$, be the consequent of the inference. $I_C \not\models E$ follows from:

1. $I_C \not\models C$ implies $I_C \not\models C'$,
2. $C \succ D$ implies $I_C \not\models D'$ (using Corollary 1), and
3. $I_C \models @_jk$ and $I_C \not\models @_k\varphi$, implies $I_C \not\models @_j\varphi$.

Lemma A7 *If $C \neq \{\}$, $I_C \not\models C$ and $dist^{S\succ}(C) = @_i\neg p$ is in reduced form, then there exists a productive clause D such that C and D may be the premises of an instance of the RES rule whose consequent is false under I_C too.*

Proof Since $I_C \not\models C$, we have $I_C \not\models @_i\neg p$ and, since it is in reduced form, it must be the case that $@_ip \in I_C$. This implies that there must exist some clause D such that $C \succ D$ and $\varepsilon_D = \{@_ip\}$ which can be, along with C, a premise of the RES rule. Since $I_C \not\models C'$ and, by Corollary 1, $I_C \not\models D'$ where $D' = D \setminus \{@_ip\}$, the consequent of such inference must be false under I_C too.

Lemma 8 (Upwards preservation for $\mathbf{R}_L^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$) *Let D be the consequent of an inference rule whose main premise is C. If $I_N \not\mathrel{|}\!\!\approx C$ and $I_C \not\mathrel{|}\!\!\approx D$, then $I_N \not\mathrel{|}\!\!\approx D$.*

Proof The structure of the proof is similar to that of the Lemma 4: from the *Main premise reduction lemma* we know $max^{\succ}(C) \succeq \varphi$ for all $\varphi \in D$ and from the hypothesis, $I_N \not\mathrel{|}\!\!\approx max^{\succ}(C)$; therefore it will suffice to show that for all φ (not necessary in D), if $max^{\succ}(C) \succ \varphi$ and $I_C \not\mathrel{|}\!\!\approx \varphi$ then $I_N \not\mathrel{|}\!\!\approx \varphi$. Again, let E be the least clause such that $I_E \not\mathrel{|}\!\!\approx \varphi$ but $I^E \mathrel{|}\!\!\approx \varphi$. Of the three possible cases, $\varepsilon_E = \{@_i p\}$ is handled exactly as in the case of $\mathbf{R}^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$, so we only have to show how to handle the other two.

First, suppose $\varepsilon_E = \{@_a b\}$ with $a, b \in \mathsf{NOM}$, which implies $a \succ b$ and $a\sigma_{I_E} = a$. In this case, φ would have to be an equality, i.e. there exist nominals i and j such that $i \succ j$, and either $\varphi = @_i j$ or $\varphi = @_j i$. In any case, we can assume $\varphi \succeq @_i j$ and, from Definition 14, $I \mathrel{|}\!\!\approx \varphi$ iff $I \mathrel{|}\!\!\approx @_i j$ for all I. Now, we will go through a rather longish case analysis that will ultimately show that it cannot be the case $a \succ i$, nor $i \succ a$, nor $a = i$, arriving, thus, to a contradiction.

1. Suppose $a \succ i$. Since $I^E \mathrel{|}\!\!\approx @_i j$, it must be the case $I^E \models @_i j$, thus, either $I_E \models @_a i$ and $I_E \models @_b j$, or $I_E \models @_a j$ and $I_E \models @_b i$. In any case, because $a \succ i$ and $a \succ j$, there must exist some k such $@_a k \in I_E$ (and $I_E \models @_k i$ or $I_E \models @_k j$). But this would contradict the fact that $a\sigma_{I_E} = a$.
2. Suppose, now, $i \succ a$. We have again two possibilities, and this time we analyze them separately.
   (a) Suppose $I_E \models @_a i$ and $I_E \models @_b j$. Since $i \succ a$, there must exist some $c \succeq a$ such that $@_i c \in I_E$ and $I_E \models @_c a$. This means that for some E' such that $E \succ E'$, $\varepsilon_{E'} = \{@_i c\}$. Now, we know, $c \succeq a \succ b$ and this implies $@_i c \succ @_a b$, which means $@_a b = max^{\succ}(E) \succ max^{\succ}(E') = @_i c \succ @_a b$
   (b) Let us assume, now, $I_E \models @_a j$ and $I_E \models @_b i$. We shall verify that it can neither be $a \succ j$ nor $j \succeq a$.
      i. If $a \succ j$ were the case, then for some k such that $a \succ k$, $@_a k \in I_E$ and $I_E \models @_k j$ should hold. But now again, this would contradict $a\sigma_{I_E} = a$.
      ii. If, on the other hand, $j \succeq a$, since $a \succ b$, we would have $@_i j \succ @_a b = max^{\succ}(E) \succeq max^{\succ}(C) \succ \varphi \succeq @_i j$.
3. Finally, assume $i = a$, which means $\varepsilon_E = \{@_i b\}$. We consider, separately, the cases $j \succeq b$ and $b \succ j$.
   (a) If $j \succeq b$, then $@_i j \succeq @_i b = max^{\succ}(E) \succeq max^{\succ}(C) \succ \varphi \succeq @_i j$.
   (b) Now suppose $b \succ j$. We have not used, so far, Condition 3 of Definition 14, but we are about to do. Since $I^E \mathrel{|}\!\!\approx @_i j$, it must be the case $I^E \models @_i j$. Now, from this and $@_i b \in I^E$, we get $I^E \models @_j b$, but since $b \succ j$, $@_i b \succ @_i j$ and, thus, we get the contradiction $I^E \not\mathrel{|}\!\!\approx @_i j$

We get to the second case. Suppose $\varepsilon_E = \{@_l \langle r \rangle k\}$; it is clear that in this case $\langle r \rangle \psi$ would have to be a formula of φ. If $\psi \notin \mathsf{NOM}$, then $\langle r \rangle \psi \succ \langle r \rangle k$ and, thus, $\varphi \succ @_l \langle r \rangle k \succeq \max(C) \succ \varphi$. Now suppose $\psi = j$ for some nominal j. If φ is of the form $@_i \langle r \rangle j$ and $I^E \mathrel{|}\!\!\approx @_i \langle r \rangle j$ because of $@_l \langle r \rangle k$, then it must be the case that $I^E \models @_i l$, $I^E \models @_j k$ and, because of Definition 14, $j \succeq k$. But since E is in weak reduced form we know $l\sigma_{I_E} = l$ and, thus, $i\sigma_{I_E} = l$, which implies $i \succeq l$. But then $\varphi = @_i \langle r \rangle j \succeq @_l \langle r \rangle k \succ max^{\succ}(C) \succ \varphi$. Finally, φ cannot be of the form $@_i \psi'$ with $\langle r \rangle j$ a proper formula of ψ' because, by Condition A4 of Definition 7, we would have $\psi' \succ \langle r \rangle k$ and, thus, $\varphi \succ @_l \langle r \rangle k \succeq max^{\succ}(C) \succ \varphi$.

Theorem 2 $\mathbf{R}_L^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ *has the reduction property for counterexamples and is, therefore, refutationally complete.*

Proof Completeness follows directly from Lemmas 7 and 8 together with Lemma A8 below.

Lemma A8 *Let $C \in N$ be such that $C \neq \{\}$ and $I_C \not\mathrel{|}\!\!\approx C$. If C is not productive, then there exists an inference in $\mathbf{R}_L^{S\succ}[\mathscr{H}^{\mathrm{NNF}}(@,\downarrow,\mathsf{A})]$ such that*

1. *C is the main premise*
2. *the side premise (if present) is productive, and*
3. *some consequent E is such that $I_C \not\mathrel{|}\!\!\approx E$*

Proof We just need to repeat the analysis done for the proof of Lemma 5. Let $\varphi = dist^{S\succ}(C)$ and let us consider every possibility:

- If $\varphi \notin \mathsf{SIMP}$, we can trivially adapt the proof of Lemma A4.
- Otherwise, if $\varphi \in \mathsf{SIMP}$,
  - If φ is not in weak reduced form, then it is not in reduced form either, and Lemma A5 can be very easily imported to this setting.
  - If, on the other hand, φ is in weak reduced form,
    - φ cannot be of in PLIT for, in that case, $S(C) = \{\}$ and, thus, C would have to be productive;
    - the case where φ is of the form $@_i \neg i$ is trivial by using the REF rule;

- if φ is of the form $@_i\neg j$ with j a nominal other than i, then it cannot be the case that $i \succ j$, for, otherwise, $I_C \not\models\!\!\approx @_i\neg j$ would imply $I_C \models @_i j$ which is only possible if for some k, $@_i k \in I_C$ and that would contradict the fact that φ is in weak reduced form;
- if $\varphi = @_i\neg j$ with $i \succ j$, a trivial counterexample is obtained using the SYM$^\neg$ rule;
- we can trivially adapt Lemma A7 for the case where φ is of the form $@_i\neg p$;
- finally, Lemma A6, trivially adapted, handles the case where φ is of the form $@_i[r]\psi$.

Theorem 6 *If* $\varphi \in \mathscr{H}^{\mathrm{NNF}}(@)$ *then:*

– $@_l\psi$ *occurs in* $\mathit{ClSet}^*_{\mathrm{S}\succ T}(\varphi)$ *implies* $\mathrm{level}(l) + d(\psi) \leq d(\varphi)$
– $@_l m$ *occurs in* $\mathit{ClSet}^*_{\mathrm{S}\succ T}(\varphi)$ *implies* $\mathrm{level}(m) \leq d(\varphi)$.

Proof First, let us define, for convenience, $d'(@_l\psi) = \mathrm{level}(l) + d(\psi)$, where $d(\psi)$ is the modal depth of φ. Now, we will proceed by induction on the number of rule applications required to derive $@_l\psi$ from the initial set $\mathit{ClSet}(\varphi)$ and we will consider separately the case where ψ is a label m, in order to verify that the second condition holds. The base case is trivially true. For the inductive step, we consider the last rule used to derive $@_l\psi$. Observe that since $\varphi \in \mathscr{H}(@)$, we don't have to consider rules A, E_ε, $\downarrow$ nor $\mathrm{PAR}^{@\Diamond}_{\text{no-}\varepsilon}$ (to see why the last one is not required, refer to the proof of Lemma 9).

Rule $\wedge$) Suppose $@_l\psi$ is generated from $@_l(\psi \wedge \psi')$ by way of the $\wedge$ rule. Then $d'(@_l\psi) \leq d'(@_l(\psi \wedge \psi')$ and, by inductive hypothesis, $d'(@_l(\psi \wedge \psi')) \leq d(\varphi)$. Furthermore, if $\psi = m$, because of Lemma 9, m cannot be an ε-term and, thus, $\mathrm{level}(m) = 0$.

Rule $\vee$) Analogous to the the $\wedge$ rule.

Rule $[r]$) Let $@_{l'}[r]\psi$ and $@_{l'}\langle r\rangle l$ be the premises of an inference using the $[r]$ rule that generates $@_l\psi$. Using the inductive hypothesis, we get $d'(@_{l'}[r]\psi) = \mathrm{level}(l') + d([r]\psi) = \mathrm{level}(l') + 1 + d(\psi) \leq d(\varphi)$. Now, by Lemma 9 we have two cases to consider:

1. $\mathrm{level}(l) = 0$ and, thus, $d'(@_l\psi) = \mathrm{level}(l) + d(\psi) = d(\psi) < d'(@_{l'}[r]\psi) \leq d(\varphi)$, or
2. $\mathrm{level}(l) = \mathrm{level}(l') + 1$, consequently, $d'(@_l\psi) = \mathrm{level}(l') + 1 + d(\psi) = d'(@_{l'}[r]\psi) \leq d(\varphi)$.

Finally, if $\psi = m$, by Lemma 9, $\mathrm{level}(m) = 0$.

Rule $\langle r\rangle_\varepsilon$) Here we have two sub-cases to consider. First, suppose from $@_{l'}\langle r\rangle\psi$ we obtain, by the $\langle r\rangle_\varepsilon$ rule, $@_l\psi$. It follows, by inductive hypothesis, that $d'(@_{l'}\langle r\rangle\psi) = \mathrm{level}(l') + d(\langle r\rangle\psi) = \mathrm{level}(l') + 1 + d(\psi) \leq d(\varphi)$. Now, since $l = \varepsilon\langle l', r, \psi\rangle$, we have $\mathrm{level}(l) = \mathrm{level}(l') + 1$, hence, $d'(@_l\psi) = \mathrm{level}(l) + d(\psi) = \mathrm{level}(l') + 1 + d(\psi) = d'(@_{l'}\langle r\rangle\psi) \leq d(\varphi)$. Observe that the side condition of the $\langle r\rangle_\varepsilon$ guarantees that $\psi \notin \mathsf{LAB}$. For the second case, suppose that from $@_l\langle r\rangle\theta$, we derive, using $\langle r\rangle_\varepsilon$ rule, $@_l\langle r\rangle\varepsilon\langle l, r, \theta\rangle$ (i.e. we are considering the case $\psi = \langle r\rangle\varepsilon\langle l, r, \theta\rangle$). Now we have $d'(@_l\langle r\rangle\varepsilon\langle l, r, \theta\rangle) = \mathrm{level}(l) + d(\langle r\rangle\varepsilon\langle l, r, \theta\rangle) = \mathrm{level}(l) + 1 \leq d'(@_l\langle r\rangle\theta) \leq d(\varphi)$.

Rule @) Suppose $@_m @_l\psi$ is the formula by which, using the @ rule we obtain $@_l\psi$. By Lemma 9, $l \in \mathsf{NOM}$ and, thus, by inductive hypothesis, $d'(@_l\psi) = d(\psi) \leq d(\varphi)$. Lemma 9 can be used again to show that if $\psi = m$, then $m \in \mathsf{NOM}$ and, thus, $\mathrm{level}(m) \leq d(\varphi)$.

Rule SYM) Suppose that $\psi = m$ and, thus, $@_l m$ is derived from $@_m l$ using the SYM rule. By inductive hypothesis, $\mathrm{level}(l) \leq d(\varphi)$ and, thus $d'(@_l m) = \mathrm{level}(l) \leq d(\varphi)$. Also by inductive hypothesis we have $\mathrm{level}(m) = d'(@_m l) \leq d(\varphi)$.

Rule SYM$^\neg$) Analogous to the SYM rule.

Rule PAR$^{@\text{no-}\Diamond}$) Let $@_{l'}\psi$ and $@_{l'}l$ be the premises from which $@_l\psi$ is obtained by way of the PAR$^{@\text{no-}\Diamond}$. We know $d'(@_{l'}\psi) = \mathrm{level}(l') + d(\psi) \leq d(\varphi)$ and, since from Definition 20 we have that $l' \succ l$ implies $\mathrm{level}(l') \geq \mathrm{level}(l)$, we conclude $d'(@_l\psi) \leq d'(@_{l'}\psi) \leq d(\varphi)$. Furthermore, if $\psi = m$, then by inductive hypothesis, $\mathrm{level}(m) \leq d(\varphi)$.

Rule PAR$^{@\Diamond}_\varepsilon$) We have to consider two cases. First, suppose $@_{l'}l$ and $@_{l'}\langle r\rangle\varepsilon\langle l', r, \theta\rangle$ are the premises from which $@_l\langle r\rangle\varepsilon\langle l, r, \theta\rangle$ is derived. By inductive hypothesis we have $d'(@_{l'}\langle r\rangle\varepsilon\langle l', r, \theta\rangle) = \mathrm{level}(l') + d(\langle r\rangle\varepsilon\langle l', r, \theta\rangle) \leq d(\varphi)$. But since it is clear that $d(\langle r\rangle\varepsilon\langle l', r, \theta\rangle) = d(\langle r\rangle\varepsilon\langle l, r, \theta\rangle)$, and $l' \succ l$ implies $\mathrm{level}(l') \geq \mathrm{level}(l)$, we conclude $d'(@_l\langle r\rangle\varepsilon\langle l, r, \theta\rangle) \leq d(\varphi)$. For the second case, suppose $@_{l'}l$ and $@_{l'}\langle r\rangle\varepsilon\langle l', r, \theta\rangle$ are the premises from which $@_{\varepsilon\langle l', r, \theta\rangle}\varepsilon\langle l, r, \theta\rangle$ is derived. On the one hand, we have $d'(@_{\varepsilon\langle l', r, \theta\rangle}\varepsilon\langle l, r, \theta\rangle) = 1 + \mathrm{level}(l') = d'(@_{l'}\langle r\rangle\varepsilon\langle l', \theta, \rangle\rangle \leq d(\varphi)$. On the other, since $l' \succ l$ implies $\mathrm{level}(l') \geq \mathrm{level}(l)$, we get $\mathrm{level}(\varepsilon\langle l, r, \theta\rangle) = 1 + \mathrm{level}(l) \leq 1 + \mathrm{level}(l') \leq d(\varphi)$.

References

1. M. Abadi and Z. Manna. Modal theorem proving. In *Proceedings of the 8th International Conference on Automated Deduction*, pages 172–189, 1986.

2. M. Abadi and Z. Manna. A timely resolution. In *Proceedings of the 1st IEEE Symposium on Logic in Computer Science*, pages 176–186, 1986.
3. R. Achmidt. A new methodology for developing deduction methods. *Annals of Mathematics and Artificial Intelligence*, 55(1–2):155–187, 2009.
4. C. Areces. *Logic Engineering. The Case of Description and Hybrid Logics*. PhD thesis, Institute for Logic, Language and Computation, University of Amsterdam, 2000.
5. C. Areces, P. Blackburn, and M. Marx. A road-map on complexity for hybrid logics. In *Proceedings of the 8th Annual Conference of the EACSL*, pages 307–321, 1999.
6. C. Areces, P. Blackburn, and M. Marx. The computational complexity of hybrid temporal logics. *Logic Journal of the IGPL*, 8(5):653–679, 2000.
7. C. Areces, H. de Nivelle, and M. de Rijke. Resolution in modal, description and hybrid logic. *Journal of Logic and Computation*, 11(5):717–736, 2001.
8. C. Areces, R. Gennari, J. Heguiabehere, and M. de Rijke. Tree-based heuristics in modal theorem proving. In W. Horn, editor, *Proceedings of ECAI 2000*, pages 199–203, 2000.
9. C. Areces and D. Gorín. Ordered resolution with selection for $\mathscr{H}(@)$. In F. Baader and A. Voronkov, editors, *Proceedings of LPAR 2004*, volume 3452 of *LNCS*, pages 125–141. Springer, 2005.
10. C. Areces and B. ten Cate. Hybrid logics. In Blackburn et al. [18], pages 821–868.
11. Y. Auffray, P. Enjalbert, and J. Hebrard. Strategies for modal resolution: results and problems. *Journal of Automated Reasoning*, 6(1):1–38, 1990.
12. F. Baader and T. Nipkow. *Term Rewriting and All That*. Cambridge University Press, 1998.
13. L. Bachmair and H. Ganzinger. Equational reasoning in saturation-based theorem proving. In *Automated deduction—a basis for applications, Vol. I*, pages 353–397. Kluwer, 1998.
14. L. Bachmair and H. Ganzinger. Resolution theorem proving. In J. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume 1, chapter 2, pages 19–99. Elsevier Science Publishers B. V., 2001.
15. P. Bieber, L. Fariñas del Cerro, and A. Herzig. MOLOG: a modal PROLOG. In *Proceedings of the 9th International Conference on Automated Deduction*, pages 762–763, 1988.
16. P. Blackburn. Representation, reasoning, and relational structures: a hybrid logic manifesto. *Logic Journal of the IGPL*, 8(3):339–365, 2000.
17. P. Blackburn, M. de Rijke, and Y. Venema. *Modal Logic*. Cambridge University Press, 2002.
18. P. Blackburn, F. Wolter, and J. van Benthem, editors. *Handbook of Modal Logics*. Elsevier, 2006.
19. M. Cialdea and L. Fariñas del Cerro. A modal Herbrand's property. *Zeitschrift fur mathematische Logik und Grundlagen der Mathematik*, 32:523–530, 1986.
20. E. Clarke, O Grumberg, and D. Peled. *Model Checking*. MIT Press, 2000.
21. H. de Nivelle and M. de Rijke. Deciding the guarded fragments by resolution. *Journal of Symbolic Computation*, 35(1):21–58, 2003.
22. H. de Nivelle, R. Schmidt, and U. Hustadt. Resolution-based methods for modal logics. *Logic Journal of the IGPL*, 8(3):265–292, 2000.
23. P. Enjalbert and L. Fariñas del Cerro. Modal resolution in clausal form. *Theoretical Computer Science*, 65(1):1–33, 1989.
24. L. Fariñas del Cerro. A simple deduction method for modal logic. *Information Processing Letters*, 14(2), 1982.
25. L. Fariñas del Cerro. Resolution modal logic. In K. Apt, editor, *Logics and Models of Concurrent Systems*, pages 27–55. Springer, 1985.
26. L. Fariñas del Cerro. MOLOG: A system that extends PROLOG with modal logic. *New Generation Computing*, 4(1), 1986.
27. M. Fitting. Destructive modal resolution. *Journal of Logic and Computation*, 1(1):83–97, 1990.
28. H. Ganzinger and H. de Nivelle. A superposition decision procedure for the guarded fragment with equality. In *LICS '99: Proceedings of the 14th Annual IEEE Symposium on Logic in Computer Science*, page 295. IEEE Computer Society, 1999.
29. M. Giese and W. Ahrendt. Hilbert's ε-terms in automated theorem proving. In N. Murray, editor, *Automated Reasoning with Analytic Tableaux and Related Methods, Intl. Conf. (TABLEAUX'99)*, volume 1617 of *LNAI*, pages 171–185. Springer, 1999.
30. E. Grädel. On the restraining power of guards. *Journal of Symbolic Logic*, 64:1719–1742, 1999.
31. E Grädel. Why are modal logics so robustly decidable? In *Current trends in theoretical computer science: entering the 21st centuary*, pages 393–408. World Scientific Publishing Co., Inc., 2001.
32. J. Herbrand. *Recherches sur la théorie de la démonstrations*. PhD thesis, Sorbone, 1930. Reprinted in W. Goldfarb, editor, *Logical Writings*. Reidel, 1971.
33. D. Hilbert and P. Bernays. *Grundlagen der Mathematik*, volume 2. Springer, 1939.
34. U. Hustadt and R. Schmidt. Using resolution for testing modal satisfiability and building models. *Journal of Automated Reasoning*, 28(2):205–232, 2002.

35. Y. Kazakov. *Saturation-Based Decision Procedures for Extensions of the Guarded Fragment*. PhD thesis, Universität des Saarlandes, 2006.
36. Y. Kazakov and B. Motik. A resolution-based decision procedure for $\mathcal{SHOIQ}$. *Journal of Automated Reasoning*, 40(2–3):89–116, 2008.
37. D. Knuth and P. Bendix. Simple word problems in universal algebras. In J. Leech, editor, *Computational Algebra*, pages 263–297. Pergamon Press, 1970.
38. A. Leisenring. *Mathematical logic and Hilbert's ε-symbol*. MacDonald, 1969.
39. S. Lindström and K. Segerberg. Modal logic and philosophy. In Blackburn et al. [18], pages 1149–1214.
40. G. Mints. Resolution calculi for modal logics. *American Mathematical Society Translations*, 143:1–14, 1989.
41. G. Mints. Gentzen-type systems and resolution rules, Part 1: Propositional logic. In *Proceedings of COLOG-88, Tallin*, volume 417 of *Lecture Notes in Computer Science*, pages 198–231. Springer, 1990.
42. C. Nalon and C. Dixon. Clausal resolution for normal modal logics. *Journal of Algorithms*, 62:117–134, 2007.
43. R. Nieuwenhuis and A Rubio. Paramodulation-based theorem proving. In A. Robinson and A. Voronkov, editors, *Handbook of Automated Reasoning*, volume 1, chapter 7, pages 371–443. Elsevier Science Publishers B. V., 2001.
44. H. Ohlbach. *A resolution calculus for modal logics*. PhD thesis, Universität Kaiserslautern, 1988.
45. A. Riazanov and A. Voronkov. Vampire 1.1 (system description). In *IJCAR '01: Proceedings of the First International Joint Conference on Automated Reasoning*, pages 376–380. Springer-Verlag, 2001.
46. A. Robinson and A. Voronkov, editors. *Handbook of automated reasoning*. Elsevier Science Publishers B. V., 2001.
47. J. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12(1):23–41, 1965.
48. R. Schmidt. Resolution is a decision procedure for many propositional modal logics. In *Advances in Modal Logic*, volume 1, pages 189–208. CSLI Publications, 1998.
49. R. Schmidt. Decidability by resolution for propositional modal logics. *Journal of Automated Reasoning*, 22(4):379–396, 1999.
50. R. Schmidt and U. Hustadt. Mechanised reasoning and model generation for extended modal logics. In H. de Swart, E. Orlowska, G. Schmidt, and M. Roubens, editors, *Theory and Applications of Relational Structures and Knowledge Instruments*, volume 2929 of *Lecture Notes in Computer Science*, pages 38–67. Springer, 2003.
51. M. Vardi. Why is modal logic so robustly decidable? In *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, volume 31, pages 149–184. AMS, 1997.
52. A. Voronkov. Algorithms, datastructures, and other issues in efficient automated deduction. In *Proceedings of IJCAR 2001*, number 2083 in LNAI, pages 13–28, 2001.
53. C. Weidenbach. System description: Spass version 1.0.0. In *CADE-16: Proceedings of the 16th International Conference on Automated Deduction*, pages 378–382. Springer-Verlag, 1999.