



Linking Focusing and Resolution with Selection

Guillaume Burel

► **To cite this version:**

| Guillaume Burel. Linking Focusing and Resolution with Selection. 2018. <hal-01670476v3>

HAL Id: hal-01670476

<https://hal.inria.fr/hal-01670476v3>

Submitted on 27 Apr 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Linking Focusing and Resolution with Selection

Guillaume Burel

ENSIIE and Samovar, Télécom SudParis and CNRS, Université Paris-Saclay, Évry, France

Inria and LSV, CNRS and ENS Paris-Saclay, Université Paris-Saclay, Cachan, France

guillaume.burel@ensiie.fr

Abstract

Focusing and selection are techniques that shrink the proof search space for respectively sequent calculi and resolution. To bring out a link between them, we generalize them both: we introduce a sequent calculus where each *occurrence* of an atom can have a positive or a negative polarity; and a resolution method where each literal, whatever its sign, can be selected in input clauses. We prove the equivalence between cut-free proofs in this sequent calculus and derivations of the empty clause in that resolution method. Such a generalization is not semi-complete in general, which allows us to consider complete instances that correspond to theories of any logical strength. We present three complete instances: first, our framework allows us to show that ordinary focusing corresponds to hyperresolution and semantic resolution; the second instance is deduction modulo theory and the related framework called superdeduction; and a new setting, not captured by any existing framework, extends deduction modulo theory with rewriting rules having several left-hand sides, which restricts even more the proof search space.

2012 ACM Subject Classification Theory of computation \rightarrow Proof theory Theory of computation \rightarrow Automated reasoning

Keywords and phrases logic in computer science, automated deduction, proof theory, sequent calculus, refinements of resolution, deduction modulo theory, polarization

Related Version <https://hal.inria.fr/hal-01670476>

1 Introduction

In addition to clever implementation techniques and data structures, a key point that explains the success of state-of-the-art automated theorem provers is the use of calculi that dramatically reduce proof search space. In the last decades, the independent developments of two families of techniques can be highlighted. First, in the kind of methods based on resolution, proof search space can be shrunk using ordering and selection techniques. The intuition is to restrict the application of the resolution rule to only some literals in a clause. If equality is considered, this leads to the superposition calculus [2] which is the base calculus of the currently most efficient automated provers for first-order classical logic. Second, in sequent calculi, Andreoli [1] introduced a technique called focusing to reduce non-determinism in the application of sequent-calculus rules. It works by first applying all invertible rules (those whose conclusion is logically equivalent to their premises) and second by chaining the application of non-invertible rules. Originally developed for linear logic, focusing has been extended to intuitionistic and classical first-order logic [28]. Focusing is mostly used in fields where sequent calculi, and related inverse and tableaux methods, are the most accurate proving method. For instance, there exists tools for first-order linear logic [13], for intuitionistic logic [29] and for modal logic [30]. Focusing is also the key ingredient in Miller's ProofCert project aiming at building a universal framework for proof certification [16].

Despite their apparent lack of relation, we show in this paper that selection in refinements of the resolution calculus and focusing in sequent calculus are in fact strongly related, so that ordinary focusing in classical first-order logic corresponds actually to hyperresolution,

45 where all negative literals are selected in a clause and are resolved at once. This connection
 46 is obtained by relaxing both techniques: concerning resolution, we allow any literal of the
 47 input clauses to be selected, whatever its sign; for the focusing part, we allow polarization
 48 not only of connectives, but also of all occurrences of literals. The main theorem of this
 49 paper, Theorem 3, shows that the sets of clauses whose insatisfiability can be proved by the
 50 resolution method with arbitrary input selection are exactly the sequents that have a cut-free
 51 proof in the generalized focusing setting.

52 This generalization allows us to cover a wider spectrum of proof systems. In particular,
 53 this permits to consider systems that search for proofs modulo some theory. Indeed, in real
 54 world applications, proof obligations are often verified within one or several theories. This
 55 explains the interest in and the success of Satisfiability Modulo Theory tools in recent years.
 56 Embedding a theory in our framework amounts to giving an axiomatic presentation of it
 57 where some literals are selected.

58 By relaxing the conditions for selecting literals, our framework is not always refutationally
 59 complete. However, this should not be considered as a drawback, but as an essential point
 60 to be able to represent efficiently all kinds of theories. Indeed, let us consider a proof search
 61 method $\mathbb{P}(\mathcal{T})$ parameterized by a theory \mathcal{T} . Ideally, $\mathbb{P}(\mathcal{T})$ should be as efficient as a generic
 62 proof search method if it is fed with a formula that is not related to the theory \mathcal{T} . In particular,
 63 if it tries to refute the true formula \top , it should terminate, and with the answer “NO”. Let
 64 us say that $\mathbb{P}(\mathcal{T})$ is relatively consistent if it is the case. As we pointed out with Dowek [9],
 65 we cannot have a generic proof of the completeness of a relatively consistent method $\mathbb{P}(\mathcal{T})$
 66 that would work for all \mathcal{T} . Indeed, such a proof would imply the consistency of the theory \mathcal{T} ,
 67 and, according to Gödel, this cannot be performed in \mathcal{T} itself. So either the completeness of
 68 the proof system is proved once and for all, but it cannot represent theories that are logically
 69 at least as strong as that proof of completeness; or it is not complete in general but it can be
 70 proved to be complete for particular theories of some arbitrary logical strength. What is
 71 interesting therefore is to give proofs of completeness of $\mathbb{P}(\mathcal{T})$ for particular theories \mathcal{T} .

Therefore, we give three instances of our framework, where we can have proofs of
 completeness. First, as stated above, we link ordinary focusing with hyperresolution, and,
 in the ground case, with semantic resolution. Second, we show that Deduction Modulo
 Theory [21] is also a particular instance of this framework, knowing that there exists numerous
 proof techniques to prove the completeness of Deduction Modulo a particular theory, for
 instance [25, 22, 19, 8]. Third, we show how completeness in our framework can be reduced
 to completeness of several instances of Deduction Modulo Theory. To give an intuition about
 this last part, and to illustrate how much the proof search space can be constrained without
 losing completeness, let us consider for example the theory defining the powerset:

$$\forall X, \forall Y, (X \in \mathcal{P}(Y)) \Leftrightarrow (\forall Z, (Z \in X) \Rightarrow (Z \in Y))$$

72 This theory can be put in clausal normal form, using d as a Skolem symbol, and we select
 73 (by underlining them) some literals in these clauses¹:

$$74 \quad \underline{\neg X \in \mathcal{P}(Y)} \vee \underline{\neg Z \in X} \vee Z \in Y \tag{1}$$

$$75 \quad \underline{X \in \mathcal{P}(Y)} \vee d(X, Y) \in X \tag{2}$$

$$76 \quad \underline{X \in \mathcal{P}(Y)} \vee \underline{\neg d(X, Y) \in Y} \tag{3}$$

¹ We use the associative-commutative-idempotent symbol \vee in clauses to distinguish it from the symbol
 \vee that is used in formulas.

78 Using focusing in general, and in our framework in particular, the decomposition of connectives
 79 is so restricted that, given an axiom, a proof derivation decomposing this axiom would
 80 necessarily have certain shapes. Thus, the axiom can be replaced by new inference rules,
 81 called synthetic rules, that are used instead of the derivation of those shapes. See end of
 82 Section 2, page 6, for more details. In our framework, this would lead to the following three
 83 synthetic rules, that can be used in place of the axioms (the explanation how these rules are
 84 obtained is given in Section 5.3):

$$\begin{array}{l}
 85 \quad (1) \vdash \frac{\Delta, u \in \mathcal{P}(v), t \in u, t \in v \vdash}{\Delta, u \in \mathcal{P}(v), t \in u \vdash} \qquad (2) \vdash \frac{}{\Delta, \neg u \in \mathcal{P}(v), d(u, v) \in v \vdash} \\
 86 \quad (3) \vdash \frac{\Delta, \neg u \in \mathcal{P}(v), d(u, v) \in u \vdash}{\Delta, \neg u \in \mathcal{P}(v) \vdash}
 \end{array}$$

87 The only proof of transitivity of the membership in the powerset is then

$$\begin{array}{l}
 87 \quad (2) \vdash \frac{a \in \mathcal{P}(b), b \in \mathcal{P}(c), \neg a \in \mathcal{P}(c), d(a, c) \in a, d(a, c) \in b, \underline{d(a, c) \in c} \vdash}{(1) \vdash \frac{a \in \mathcal{P}(b), b \in \mathcal{P}(c), \neg a \in \mathcal{P}(c), d(a, c) \in a, \underline{d(a, c) \in b} \vdash}{(1) \vdash \frac{a \in \mathcal{P}(b), b \in \mathcal{P}(c), \neg a \in \mathcal{P}(c), \underline{d(a, c) \in a} \vdash}{(3) \vdash \frac{a \in \mathcal{P}(b), b \in \mathcal{P}(c), \neg a \in \mathcal{P}(c) \vdash}{\wedge \vdash \frac{a \in \mathcal{P}(b) \wedge b \in \mathcal{P}(c) \wedge \neg a \in \mathcal{P}(c) \vdash}{\exists \vdash \frac{\exists A. \exists B. \exists C. A \in \mathcal{P}(B) \wedge B \in \mathcal{P}(C) \wedge \neg A \in \mathcal{P}(C) \vdash}}}}}}
 \end{array}$$

89 where the active formulas in a sequent are underwaved, and double lines indicate potentially
 90 several applications of an inference rule.

91 On the resolution side, clauses (1) to (3) lead to the following ground derived rules (see
 92 also Section 5.3):

$$93 \quad (1) \frac{u \in \mathcal{P}(v) \uparrow C \quad t \in u \uparrow D}{t \in v \uparrow C \uparrow D} \quad (2) \frac{\neg u \in \mathcal{P}(v) \uparrow C}{d(u, v) \in u \uparrow C} \quad (3) \frac{\neg u \in \mathcal{P}(v) \uparrow C \quad d(u, v) \in v \uparrow D}{C \uparrow D}$$

94 Once again, there is only one proof of transitivity, i.e. starting from the set of clauses
 95 $\{a \in \mathcal{P}(b); b \in \mathcal{P}(c); \neg a \in \mathcal{P}(c)\}$:

$$96 \quad (3) \frac{\neg a \in \mathcal{P}(c) \quad (1) \frac{b \in \mathcal{P}(c) \quad (1) \frac{a \in \mathcal{P}(b) \quad (2) \frac{\neg a \in \mathcal{P}(c)}{d(a, c) \in a}}{d(a, c) \in b}}{d(a, c) \in c}}{\square}$$

97 and we cannot even infer other clauses than those. We let the reader compare with what
 98 happens if we used clauses (1) to (3) in resolution, even using the ordered resolution with
 99 selection refinement.

100 **Related work.** Chaudhuri et al. [14] show that hyperresolution for Horn clauses can be
 101 explained as an instance of a sequent calculus for intuitionistic linear logic with focusing
 102 where atoms are given a negative polarity.

103 Farooque et al. [24] developed a sequent calculus, based on focusing, that is able to
 104 simulate DPLL(\mathcal{T}), the most common calculus used in SMT provers. The main difference
 105 with our framework is that in [24], the theory is considered as a black box which is called as
 106 an oracle. Here, the theory is considered as a first-class citizen.

107 Within the ProofCert project, resolution proofs can be checked by a kernel built upon
 108 a sequent calculus with focusing [16]. Based on this, the tool `Checkers` [15] is able to

109 verify proofs coming from automated theorem provers based on resolution such as E-prover.
 110 Different from here, they translate resolution derivations using cuts to get smaller proofs.

111 Hermant [26] proves the correspondance between the cut-free fragment of a sequent
 112 calculus and a resolution method, in the setting of Deduction Modulo Theory. Since
 113 Deduction Modulo Theory is subsumed by our framework, Theorem 3 is a generalization of
 114 Hermant’s work. Proving it is simpler in our setting because focusing restrains the shape of
 115 possible sequent calculus proofs, whereas Hermant had to prove technical lemmas to give
 116 proofs a canonical shape.

117 **Notations and conventions.** We use standard definitions for terms, predicates, formulas
 118 (with connectives $\perp, \top, \neg, \wedge, \vee$ and quantifiers \forall, \exists), sequents and substitutions. A literal
 119 is an atom or its negation. A clause is a set of literals. We will identify a literal with the
 120 unit clause containing it. Unless stated otherwise, letters P, Q, R, P', P_1, \dots denote atoms,
 121 L, K, L', L_1, \dots denote literals, A, B, A', A_1, \dots denote formulas, C, D, C', C_1, \dots denote
 122 clauses, Γ, Δ denote set of clauses or set of formulas (depending on the context). A^\perp denotes
 123 the negation normal form of $\neg A$.

124 **2 Focusing with Polarized Occurrences of Atoms**

125 Focusing was introduced by Andreoli [1] to restrict the non-determinism in some sequent
 126 calculus for linear logic. It relies on the alternation of two phases: During the asynchronous
 127 phase (sequents with \uparrow), all invertible rules are applied on the formulas of the sequent. Recall
 128 that a rule is said invertible if its conclusion implies the conjunction of its premises. During
 129 the synchronous phase (sequents with \downarrow), a particular formula is selected —the focus is on
 130 it— and all possible non-invertible rules are successively applied on it. This idea has been
 131 extended to intuitionistic and classical first-order logic [28]. In these, connectives may have
 132 invertible and non-invertible versions of their sequent calculus rules. Therefore, one considers
 133 in that case two versions of a connective, one called positive when the right introduction
 134 rule is non-invertible, and one called negative when it is invertible. Some connectives, i.e.
 135 \exists in classical logic, only have a positive version, and dually, others, such as \forall in classical
 136 logic, only have a negative version. Given a usual formula, one can decide which version of
 137 a connective one wants to use at a particular occurrence, which is called a polarization of
 138 the formula.² Note that the polarity of a connective does not affect its semantics, it only
 139 alters the shape of the sequent calculus proofs. Similarly, one can decide the polarity of
 140 each literal. If a literal with negative polarity L is focused on in a branch, then this branch
 141 must necessarily be closed, with L^\perp in the same context. (See rule $\widehat{\downarrow} \vdash$ in Figure 1.) In the
 142 ordinary presentation of focusing, this polarity is chosen globally for all occurrences of each
 143 atom, and the polarity of $\neg P$ is defined as the inverse of that of P . In our setting, the polarity
 144 is attached to the position of the literal in the formula. In particular, if a substitution is
 145 applied to the formula, the polarities of the resulting literals do not change. The polarity
 146 of a formula is defined as the polarity of its top connective. Besides, note that to switch
 147 the polarity of a formula, e.g. to impose a change of phase, one can prefix it by so-called
 148 delays: $\delta^- A$ is negative whatever the polarity of A . Delays can be defined for instance by
 149 $\delta^- A = \forall x. A$ where x is not free in A , so we do not need them in the syntax and the rules.

² Let us note that this notion of polarity is a standard denomination when dealing with focusing, and should not be confused with the more usual but unrelated notion defined by the parity of the negation-depth of a position in a formula.

$$\begin{array}{c}
\text{Asynchronous phase:} \quad \widehat{\uparrow} \vdash \frac{}{\Gamma, L, L^\perp \uparrow \vdash} \quad \uparrow \exists \vdash \frac{\Gamma \uparrow \Delta, A \vdash}{\Gamma \uparrow \Delta, \exists x. A \vdash} \quad x \text{ not free in } \Gamma, \Delta \\
\uparrow \vee \vdash \frac{\Gamma \uparrow \Delta, A \vdash \quad \Gamma \uparrow \Delta, B \vdash}{\Gamma \uparrow \Delta, A \vee^+ B \vdash} \quad \uparrow \wedge \vdash \frac{\Gamma \uparrow \Delta, A, B \vdash}{\Gamma \uparrow \Delta, A \wedge^+ B \vdash} \quad \uparrow \top \vdash \frac{\Gamma \uparrow \Delta \vdash}{\Gamma \uparrow \Delta, \top \vdash} \\
\hline
\text{Synchronous phase:} \quad \widehat{\downarrow} \vdash \frac{}{\Gamma, L^\perp \downarrow \underline{L} \vdash} \quad \downarrow \forall \vdash \frac{\Gamma \downarrow \{t/x\} A \vdash}{\Gamma \downarrow \forall x. A \vdash} \quad \downarrow \perp \vdash \frac{}{\Gamma \downarrow \perp \vdash} \\
\downarrow \vee \vdash \frac{\Gamma \downarrow A \vdash \quad \Gamma \downarrow B \vdash}{\Gamma \downarrow A \vee^- B \vdash} \quad \downarrow \wedge_1 \vdash \frac{\Gamma \downarrow A \vdash}{\Gamma \downarrow A \wedge^- B \vdash} \quad \downarrow \wedge_2 \vdash \frac{\Gamma \downarrow B \vdash}{\Gamma \downarrow A \wedge^- B \vdash} \\
\hline
\text{Focus} \quad \frac{\Gamma, A \downarrow A \vdash}{\Gamma, A \uparrow \vdash} \quad \text{Release} \quad \frac{\Gamma \uparrow A \vdash}{\Gamma \downarrow A \vdash} \quad \text{Store} \quad \frac{\Gamma, A \uparrow \Delta \vdash}{\Gamma \uparrow A, \Delta \vdash} \\
A \text{ negative} \quad \quad \quad A \text{ positive} \quad \quad \quad A \text{ negative or literal}
\end{array}$$

■ **Figure 1** The sequent calculus LKF^\perp

150 Liang and Miller [28] introduce the sequent calculus LKF, and prove it to be complete
 151 for classical first-order logic. In Figure 1, we present the calculus LKF^\perp , which is almost the
 152 same with the following differences:

- 153 ■ All formulas are put on the left-hand side of the sequent, instead of the right-hand side.
 154 Therefore, one does not try to prove a disjunction of formulas, but one tries to refute a
 155 conjunction of formulas. This is the same thanks to the dual nature of classical first-order
 156 logic, and this helps to be closer to the resolution derivations. Note that, consequently,
 157 the focus is on negative formulas, and invertible rules are applied on positive formulas.
- 158 ■ The polarity of atoms is not chosen globally, but each *occurrence* of a literal can have a
 159 positive or a negative polarity. In particular, we can have two literals L and L^\perp which
 160 are both negative, or both positive. We denote by \underline{L} the fact that the literal L has a
 161 negative polarity. To be able to close branches on which we have two positive opposed
 162 literals, we add a rule $\widehat{\uparrow} \vdash$.

163 We denote by $\Gamma \uparrow \Delta \vdash$ (with Γ or Δ , possibly empty, containing polarized formulas)
 164 the fact that there exists a proof of the sequent $\Gamma \uparrow \Delta \vdash$ in LKF^\perp , that is, a derivation
 165 starting from this sequent and whose branches are all closed (by $\widehat{\downarrow} \vdash$, $\widehat{\uparrow} \vdash$ or $\downarrow \perp \vdash$). Thanks
 166 to focusing, such a proof has the following shape :

- 167 ■ Since one starts in an asynchronous (\uparrow) phase, invertible rules are successively applied to
 168 the positive formulas of Δ , until one obtains negative formulas or literals that are put on
 169 the left of \uparrow using **Store**.
 - 170 ■ When no formula appears on the right of \uparrow , then either the branch is closed by $\widehat{\uparrow} \vdash$; or
 171 the focus is put on a negative formula using **Focus**.
 - 172 ■ In the latter case, one is now in synchronous (\downarrow) phase where non-invertible rules are
 173 successively applied to the formula upon which the focus is, until either the branch is
 174 closed using $\widehat{\downarrow} \vdash$ or $\downarrow \perp \vdash$; or one obtains a positive formula and the synchronous phase
 175 ends using **Release**.
 - 176 ■ In the latter case, one starts again in the asynchronous phase.
- 177 Focusing therefore strongly constraints the shape of possible proofs, and therefore reduces
 178 the proof search space. The $\widehat{\downarrow} \vdash$ in particular imposes to close branches immediately when
 179 the focus is on a negative literal, and thus rules out many derivations.

180 Note that proofs can be closed when the polarities of an atom and its negation are both
 181 positive (rule $\widehat{\uparrow} \vdash$), or when one is positive and the other negative (rule $\widehat{\downarrow} \vdash$), but not when

182 they are both negative. Therefore, this restricts how formulas that contains literals with
 183 negative polarities can interact one with the others, and this is the main point of LKF^\perp to
 184 reduce the proof search space.

185 Restricting proof search using focusing leads to what are called synthetic rules (see for
 186 instance [14, pp.148–150] where they are called derived rules). The idea is to replace some
 187 formula A in the context of the sequent by new inference rules. Instead of proving the sequent
 188 $A, \Delta \vdash$ in LKF^\perp , one proves $\Delta \vdash$ in $(\text{LKF}^\perp + \text{the synthetic rules obtained from } A)$. Indeed,
 189 a proof focusing on A can only have certain shapes, and thus instead of having A in the
 190 context, it can be replaced by new rules synthesizing those shapes. For instance, the formula
 191 $\underline{P} \vee^- (Q \wedge^+ \underline{R})$ in a context Γ can only lead to the following derivations when the focus is
 192 put on it:

$$\begin{array}{c}
 \widehat{\Downarrow} \vdash \frac{}{\Gamma \Downarrow P \vdash} \quad \Downarrow \wedge_1 \vdash \frac{}{\Gamma \Downarrow Q \wedge^- \underline{R} \vdash} \quad \text{Store} \frac{\Gamma, Q \uparrow \vdash}{\Gamma \uparrow Q \vdash} \quad \text{Release} \frac{\Gamma \uparrow Q \vdash}{\Gamma \Downarrow Q \vdash} \\
 \Downarrow \vee \vdash \frac{}{\Gamma \Downarrow P \vee^- (Q \wedge^+ \underline{R}) \vdash} \quad \text{Focus} \frac{}{\Gamma \uparrow \vdash} \quad \text{and} \quad \widehat{\Downarrow} \vdash \frac{}{\Gamma \Downarrow P \vdash} \quad \Downarrow \wedge_2 \vdash \frac{\widehat{\Downarrow} \vdash \frac{}{\Gamma \Downarrow \underline{R} \vdash}}{\Gamma \Downarrow Q \wedge^- \underline{R} \vdash} \\
 \Downarrow \vee \vdash \frac{}{\Gamma \Downarrow P \vee^- (Q \wedge^+ \underline{R}) \vdash} \quad \text{Focus} \frac{}{\Gamma \uparrow \vdash}
 \end{array}$$

194 In the left derivation, P^\perp must be in Γ to be able to close the left branch, so Γ is in fact of the
 195 form $\underline{P} \vee^- (Q \wedge^+ \underline{R}), \Delta, P^\perp$. In the right one, Γ must be of the form $\underline{P} \vee^- (Q \wedge^+ \underline{R}), \Delta, P^\perp, R^\perp$.
 196 Instead of searching for a proof with $\underline{P} \vee^- (Q \wedge^+ \underline{R})$ in the context, the following two synthetic
 197 rules can therefore be used:

$$\text{Syn1} \frac{\Delta, P^\perp, Q \uparrow \vdash}{\Delta, P^\perp \uparrow \vdash} \quad \text{Syn2} \frac{}{\Delta, P^\perp, R^\perp \uparrow \vdash}$$

199 Provability is the same because each application of a synthetic rule can be replaced by
 200 applying Focus on $\underline{P} \vee^- (Q \wedge^+ \underline{R})$ and following the derivation leading the synthetic rule, and
 201 vice versa. This is used for instance in provers based on the inverse method and focusing [29].

202 The sequent calculus LKF^\perp is not complete in general. One of the simplest examples
 203 of incompleteness is the sequent $\underline{P} \vee^- Q, \neg \underline{P} \vee^- Q, \neg Q \uparrow \vdash$ which has no proof although
 204 $P \vee Q, \neg P \vee Q, \neg Q$ is not satisfiable.

205 **3** Resolution with Input Selection

206 Two approaches can be used to reduce the proof search space of the resolution calculus: first,
 207 one can restrict on which pairs of clauses the resolution rule can be applied; this leads for
 208 instance to the set-of-support strategy [34], in which clauses are split into two sets, called the
 209 theory and the set of support; at least one of the clauses involved in a resolution step must
 210 be in the set of support. Second, one can restrict which literals in the clauses can be resolved
 211 upon; those literals are said to be selected in the clause. Resolution with free selection is
 212 complete for Horn clauses, but incomplete in general. Selecting a subset of the negative
 213 literals (if no literal is selected, then any literal of the clause can be used in resolution) is
 214 however complete, and combining this with an ordering restriction on clauses with no selected
 215 literals leads to Ordered Resolution with Selection, which was introduced by Bachmair and
 216 Ganzinger [2] (see also [3]) as a complete refinement of resolution.

217 Resolution with Input Selection combines these two approaches. It is parameterized by a
 218 selection function \mathcal{S} that associate to each input clause a subset of its literals. If the selection
 219 function selects at least one literal, only those can be used in Resolution. Otherwise, any
 220 of them can be used. Note that for generated clauses, we impose that $\mathcal{S}(C) = \emptyset$. We also
 221 allow to have the same input clause several times with different selections. (That is, we

$$\begin{array}{l}
\text{Resolution } \frac{L \vee C \quad \underline{L'}^\perp \vee D}{\sigma(C \vee D)} \qquad \text{Factoring } \frac{L \vee L' \vee C}{\sigma(L \vee C)} \\
\blacksquare \mathcal{S}(L \vee C) = \emptyset \qquad \blacksquare \mathcal{S}(L \vee L' \vee C) = \emptyset \\
\blacksquare \mathcal{S}(\underline{L'}^\perp \vee D) = \emptyset \qquad \blacksquare \sigma \text{ is the most general unifier of } L =? L' \\
\blacksquare \sigma \text{ is the most general unifier of } L =? L' \\
\\
\text{Resolution with Selection } \frac{\underline{K_1} \vee \dots \vee \underline{K_n} \vee C \quad \underline{K'_1}^\perp \vee D_1 \quad \dots \quad \underline{K'_n}^\perp \vee D_n}{\sigma(C \vee D_1 \vee \dots \vee D_n)} \\
\blacksquare \mathcal{S}(\underline{K_1} \vee \dots \vee \underline{K_n} \vee C) = \{K_1; \dots; K_n\} \\
\blacksquare \mathcal{S}(\underline{K'_i}^\perp \vee D_i) = \emptyset \\
\blacksquare \sigma \text{ is the mgu of the simultaneous unification problem } K_1 =? K'_1, \dots, K_n =? K'_n
\end{array}$$

■ **Figure 2** Resolution with Input Selection

222 actually work with couples composed of a clause and its selected literals.) The inference
223 rules of Resolution with Input Selection are presented in Fig. 2. Literals that are selected in
224 a clause are underlined. We will see that they indeed correspond to the literals that have a
225 negative polarization in LKF^\perp . As usual, variables are renamed in the clauses to avoid that
226 premises of the inference rules share variables. We have two flavors of the resolution rule:
227 the usual binary resolution, that is applied on two premises that do not select any literal;
228 and Resolution with Selection that is applied on a clause in which n literals are selected and
229 n clauses in which no literal is selected. Consequently, clauses with a non-empty selection
230 cannot be resolved one with the others. By considering them as the theory part, and the
231 clauses with an empty selection as the set of support, it is easy to see that Resolution with
232 Input Selection is a generalization of the set-of-support strategy. Notwithstanding, note
233 that neither Resolution with Input Selection is a generalization of Ordered Resolution with
234 Selection nor the converse.

235 ► **Definition 1** (Resolution derivation). We write $\Gamma \rightsquigarrow C$ if C can be derived from some clauses
236 in Γ using the inference rules Resolution with Selection, Resolution, or Factoring presented in
237 Figure 2. We write $\Gamma \rightsquigarrow^* C$ if
238 ■ $C \in \Gamma$ or if
239 ■ there exists D such that $\Gamma \rightsquigarrow D$ and $\Gamma, D \rightsquigarrow^* C$.

240 As usual in resolution methods, the goal is to produce the empty clause \square starting from a
241 set of clauses Γ to show, since all rules are sound, that Γ is unsatisfiable. Here again, the
242 calculus is not complete in general: from the set of clauses $\underline{P} \vee Q, \underline{\neg P} \vee Q, \neg Q$, no inference
243 rule can be applied: to apply Resolution with Selection, we would need a clause where P , or
244 $\neg P$, is not selected, and Resolution needs two clauses without selection.

245 **4** LKF^\perp is a Conservative Extension of Resolution with Input 246 Selection

247 To link LKF^\perp with Resolution with Input Selection, we need to indicate how clauses are
248 related to polarized formulas.

249 ► **Definition 2.** Given a clause $C = \underline{L_1} \vee \dots \vee \underline{L_n} \vee K_1 \vee \dots \vee K_m$ whose free variables are
250 x_1, \dots, x_l and such that $\mathcal{S}(C) = \{L_1; \dots; L_n\}$, we define the associated formula $\lceil C \rceil =$
251 $\forall x_1, \dots, x_l. \underline{L_1} \vee^- \dots \vee^- \underline{L_n} \vee^- \delta^-(K_1 \vee^+ \dots \vee^+ K_m)$. $\lceil C \rceil$ is said to be in clausal form.
252 By extension, $\lceil \Gamma \rceil$ is the set of the formulas associated to the clauses of the set Γ .

253 The main theorem of this article relates LKF^\perp with Resolution with Input Selection:

254 ► **Theorem 3.** *Let Γ be a set of clauses. We have $\lceil \Gamma \rceil \uparrow \vdash$ iff $\Gamma \rightsquigarrow^* \square$.*

255 The proof is given in the appendix. To prove the right-to-left direction, we prove that
 256 all inference rules of Resolution with Input Selection are admissible in LKF^\perp , in the sense
 257 that if $\Gamma \rightsquigarrow C$ then LKF^\perp proofs of $\lceil \Gamma \rceil, \lceil C \rceil \uparrow \vdash$ can be turned into proofs of $\lceil \Gamma \rceil \uparrow \vdash$. Note
 258 that they are admissible, but they are not derivable. In particular, the size of the proof in
 259 LKF^\perp can be much larger than the resolution derivation, as expected in a cut-free sequent
 260 calculus. Using cuts would lead to a closer correspondence between resolution derivations
 261 and sequent-calculus proofs, as in [16]. However, we chose to stay in the cut-free fragment to
 262 prove that, even in the incomplete case, resolution coincides with cut-free proofs, as in [26].

263 5 Complete Instances

264 5.1 Ordinary Focusing and Semantic Hyperresolution

265 As said earlier, in standard LKF, not all occurrences of literals can have an arbitrary polarity.
 266 Instead, each atom P is given globally a polarity, and P^\perp has the opposite polarity.

267 Let us first look at the simple case where atoms are given a positive polarity. We recall
 268 the completeness proof of LKF:

269 ► **Theorem 4** (Corollary of [28, Theorem 17]). *If the literals with a positive polarity are
 270 exactly the atoms, LKF^\perp is (sound and) complete.*

271 If we look at the corresponding resolution calculus, Resolution with Selection for this particular
 272 instance becomes:

$$273 \text{R.w.S.} \frac{\neg P_1 \vee \dots \vee \neg P_n \vee C \quad P'_1 \vee D_1 \quad \dots \quad P'_n \vee D_n}{\sigma(C \vee D_1 \vee \dots \vee D_n)}$$

274 where C and D_i for all i contain only positive literals, and σ is the most general unifier of
 275 $P_1 =? P'_1, \dots, P_n =? P'_n$. Note that the clause $\sigma(C \vee D_1 \vee \dots \vee D_n)$ contains only positive
 276 literals, so no literal would be selected in it even if it was an input clause. Besides, Resolution
 277 cannot be applied, since there exists no clause $\neg P \vee C$ with $\mathcal{S}(\neg P \vee C) = \emptyset$.

278 This corresponding resolution calculus is therefore exactly hyperresolution of [31]: premises
 279 of an inference contains all only positive literals, except one clause whose all negative literals
 280 are resolved at once. Theorem 3 therefore links ordinary focusing with hyperresolution.
 281 Consequently, Theorem 4 implies the completeness of hyperresolution.

282 Chaudhuri et al. [14, Theorem 16] prove a similar result by establishing a correspondence
 283 between hyperresolution derivations and proofs in a focused sequent calculus for intuitionistic
 284 linear logic, but only considering Horn clauses. In their setting, choosing a negative polarity
 285 for atoms leads to SLD resolution, which is the reasoning mechanism of Prolog.

286 Let us now look at the general case, where atoms are given an arbitrary polarity. Let
 287 us stick to the ground case. We first recall a refinement of resolution called Semantic
 288 hyperresolution [33][12, Sect. 1.3.5.3]. Let I be an arbitrary Herbrand interpretation, i.e.
 289 a model whose domain is the set of terms interpreted as themselves. Note that I is not
 290 assumed to be a model of the input set of clauses (which is fortunate, since one is trying
 291 to show that it is unsatisfiable). Given a clause C , the idea of semantic hyperresolution is
 292 to resolve all literals of C that are valid in I at once, with clauses whose literals are all not
 293 valid in I . This gives the rule:

$$\text{SHR} \frac{K_1 \curlywedge \dots \curlywedge K_n \curlywedge C \quad K_1^\perp \curlywedge D_1 \quad \dots \quad K_n^\perp \curlywedge D_n}{C \curlywedge D_1 \curlywedge \dots \curlywedge D_n}$$

where for all i , $I \models K_i$ (and thus $I \not\models K_i^\perp$), $I \not\models C$ and $I \not\models D_i$. Note that $I \not\models C \curlywedge D_1 \curlywedge \dots \curlywedge D_n$.

Semantic hyperresolution for a Herbrand interpretation I can be seen as an instance of Resolution with Input Selection by using the following polarization of atoms: a literal L has a negative polarity iff $I \models L$. In that case, SHR corresponds exactly to Resolution with Selection, and Resolution cannot be applied since we cannot have clauses $P \curlywedge C$ and $\neg P \curlywedge D$ where both P and $\neg P$ are not valid in I .

This particular instance of polarization is in fact the ordinary version of focusing. Indeed, once a global polarity is assigned to each atom, the set of literals whose polarity is negative defines an Herbrand interpretation, and we saw reciprocally how to design a global polarization from the Herbrand interpretation. Theorem 3 therefore links ordinary focusing in the ground case with semantic hyperresolution. They are both complete, thanks to this theorem:

► **Theorem 5** (Corollary of [28, Theorem 17]). *Given a global polarization of atoms, where the polarity of P^\perp is the opposite of that of P , LKF^\perp is (sound and) complete.*

5.2 Deduction Modulo Theory

Deduction Modulo Theory [21] is a framework that consists in applying the inference rules of an existing proof system modulo some congruence over formulas. This congruence represents the theory, and it is in general defined by means of rewriting rules. To be expressive enough, these rules are defined not only at the term level, but also for formulas. To get simpler presentations of theories, we distinguish between rewrite rules that can be applied at positive and at negative positions by giving them a polarity³, where by negative position we mean under an odd number of \neg . We therefore have positive rules $P \rightarrow^+ A$ and negative rules $P \rightarrow^- A$ where P is an atom and A an arbitrary formula whose free variables appears in P . Given a rule $P \rightarrow^+ A$, the rewrite relation $B_1 \xrightarrow{+} B_2$ is defined as usual by saying that there exists a position \mathfrak{p} and a substitution σ such that the subformula of B_1 at position \mathfrak{p} is σP and B_2 equals B_1 where the subformula at position \mathfrak{p} is replaced by σA . $\xrightarrow{-}$ is defined similarly. In Polarized Sequent Calculus Modulo theory [18], the inference rules of the sequent calculus are applied modulo such a polarized rewriting system, as in for instance in $\vdash \wedge \frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash C, \Delta} C \xrightarrow{+} *A \wedge B$. Note that the implicit semantics of a negative rule $P \rightarrow^- A$ is therefore $\overline{\forall x}.$ ($P \Rightarrow A$), whereas the semantics of $P \rightarrow^+ A$ is $\overline{\forall x}.$ ($A \Rightarrow P$), where \overline{x} are the free variables of P .

With Kirchner [10], we proved the equivalence of Polarized Sequent Calculus Modulo theory to a sequent calculus where polarized rewriting rules are applied only on literals, using explicit rules. This calculus, Polarized Unfolding Sequent Calculus, is almost the calculus PUSC[⊥] presented in Figure 3. The only difference is that all formulas are put on the left of the sequent in PUSC[⊥]. We denote by $\Gamma \vdash_{\mathcal{R}}$ the fact that $\Gamma \vdash$ can be proved in PUSC[⊥] using the polarized rewriting system \mathcal{R} . Note that the rule for the universal quantifier $\forall \vdash$ as well as the unfolding rules $\uparrow^- \vdash$ and $\uparrow^+ \vdash$ contain an implicit contraction rule, as in the sequent calculus G4 of Kleene, in order to ensure that all rules of PUSC[⊥] are invertible.

We can translate polarized rewriting rules as formulas with selection, and see PUSC[⊥] as an instance of LKF[⊥]. We first consider how to translate formulas of the right-hand side of

³ This polarity must not be confused with the other notions of polarity mentioned in the paper.

23:10 Linking Focusing and Resolution with Selection

$$\begin{array}{c}
\text{⊢} \frac{}{\Gamma, L, L^\perp \vdash} \quad \text{⊤} \frac{\Gamma \vdash}{\Gamma, \top \vdash} \quad \text{⊥} \frac{}{\Gamma, \perp \vdash} \quad \text{⋁} \frac{\Gamma, A \vdash \quad \Gamma, B \vdash}{\Gamma, A \vee B \vdash} \\
\text{⋀} \frac{\Gamma, A, B \vdash}{\Gamma, A \wedge B \vdash} \quad \text{⋔} \frac{\Gamma, A \vdash}{\Gamma, \exists x. A \vdash} \quad x \text{ not free in } \Gamma \quad \text{⋄} \frac{\Gamma, \forall x. A, \{t/x\}A \vdash}{\Gamma, \forall x. A \vdash} \\
\text{⋈}^- \frac{\Gamma, P, A \vdash}{\Gamma, P \vdash} P \multimap^- A \quad \text{⋈}^+ \frac{\Gamma, \neg P, A^\perp \vdash}{\Gamma, \neg P \vdash} P \multimap^+ A
\end{array}$$

■ **Figure 3** The sequent calculus PUSC^\perp

$$\begin{array}{c}
\text{Resolution} \frac{P \Upsilon C \quad \neg Q \Upsilon D}{\sigma(C \Upsilon D)} \quad \text{Factoring} \frac{L \Upsilon K \Upsilon C}{\sigma(L \Upsilon C)} \quad \sigma = \text{mgu}(L, K) \\
\text{Ext. Narr.}^- \frac{P \Upsilon C}{\sigma(D \Upsilon C)} \quad \text{Ext. Narr.}^+ \frac{\neg Q \Upsilon D}{\sigma(C \Upsilon D)} \quad \text{Ext. Narr.}^+ \frac{\neg Q \Upsilon D}{\sigma(C \Upsilon D)} \quad \sigma = \text{mgu}(P, Q)
\end{array}$$

^a $\sigma = \text{mgu}(P, Q)$

■ **Figure 4** Inference rules of Polarized Resolution Modulo theory

polarized rewriting rules. We polarize them by choosing positive connectives for \vee and \wedge and, to unchain the introduction of the universal quantifier, we introduce delays. (Let us recall that a delay δ^+ allows to force a formula to be positive, and it can be encoded using an existential quantifier.) This gives the translation:

$$\begin{array}{c}
|L| = L \quad \text{when } L \text{ is } \top, \perp \text{ or a literal} \quad |A \wedge B| = |A| \wedge^+ |B| \\
|A \vee B| = |A| \vee^+ |B| \quad |\exists x. A| = \exists x. |A| \quad |\forall x. A| = \forall x. \delta^+ |A|
\end{array}$$

334 ► **Definition 6.** Given a negative rewriting rule $P \rightarrow^- A$ where the free variables of P are
335 x_1, \dots, x_n , its translation as a formula with selection is $\llbracket P \rightarrow^- A \rrbracket = \forall x_1 \dots x_n. \underline{\neg P} \vee^- \delta^+ |A|$.

336 Given a positive rewriting rule $P \rightarrow^+ A$ where the free variables of P are x_1, \dots, x_n , its
337 translation as a formula with selection is $\llbracket P \rightarrow^+ A \rrbracket = \forall x_1. \dots \forall x_n. \underline{P} \vee^- \delta^+ |A^\perp|$.

338 The translation $\llbracket \mathcal{R} \rrbracket$ of a polarized rewriting system \mathcal{R} is the multiset of the translation
339 of its rules.

340 ► **Definition 7.** Let N_1, \dots, N_n be a multiset of formulas whose top connective is \forall or \perp or
341 that are literals, and let P_1, \dots, P_m be a multiset of non-literal formulas whose top connective
342 is neither \forall nor \perp , then the translation of the PUSC^\perp sequent $N_1, \dots, N_n, P_1, \dots, P_m \vdash$
343 modulo the rewriting system \mathcal{R} is the LKF^\perp sequent $\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, |N_n| \uparrow |P_1|, \dots, |P_m| \vdash$.

344 ► **Theorem 8.** $N_1, \dots, N_n, P_1, \dots, P_m \vdash_{\mathcal{R}}$ in PUSC^\perp iff $\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, |N_n| \uparrow |P_1|, \dots, |P_m| \vdash$
345 in LKF^\perp .

346 The proof is given in the appendix.

347 Let us now consider the subcase where the rewriting rules are clausal, according to the
348 terminology of [20], e.g. they are of the form $P \rightarrow^- C$ or $P \rightarrow^+ \neg C$ for some formula C
349 in clausal normal form. In that case, the resolution method based on Deduction Modulo
350 Theory [21] can be refined into what is called Polarized Resolution Modulo theory [20], whose
351 rules are given in Fig. 4. (A refinement of) Polarized Resolution Modulo theory is actually
352 implemented in the automated theorem prover *iProverModulo* [6].

353 By noting that the translation of the rule $Q \rightarrow^- D$ is $\llbracket Q \rightarrow^- D \rrbracket = \forall x_1. \dots \forall x_n. \underline{\neg Q} \vee^-$
354 $\delta^+ |D|$ whereas $\ulcorner \underline{\neg Q} \Upsilon D \urcorner = \forall x_1. \dots \forall x_n. \underline{\neg Q} \vee^- \delta^- |D|$, we can relate the rule $Q \rightarrow^- D$
355 with the clause with selection $\underline{\neg Q} \Upsilon D$, which is called a one-way clause by [20]. Indeed,

356 the change of phase is always needed in that particular case, so that the delays are in fact
 357 useless. Ext. Narr.^- can therefore be seen as an instance of the Resolution with Selection
 358 rule: Resolution with Selection $\frac{\neg Q \vee D \quad P \vee C}{\sigma(D \vee C)} \sigma = \text{mgu}(P, Q)$. Similarly, $P \rightarrow^+ \neg C$ is
 359 related to $\underline{P} \vee C$.

360 Consequently, since PUSC^\perp corresponds to LKF^\perp , and Resolution with Input Selection
 361 corresponds to Polarized Resolution Modulo theory, Theorem 3 leads to a new and more
 362 generic proof of the correspondence between PUSC^\perp and Polarized Resolution Modulo theory.

363 Deduction Modulo Theory is not always complete. This is the case only if the cut rule
 364 is admissible in Polarized Sequent Calculus Modulo theory. It holds for some particular
 365 theories, e.g. Simple Type Theory [21] and arithmetic [23]. There are more or less powerful
 366 techniques that ensures this property [25, 22, 19, 8]. We even proved that any consistent
 367 first-order theory can be presented by a rewriting system admitting the cut rule [7]. As
 368 presented with Dowek [9] and discussed in the introduction, the fact that completeness is not
 369 proved once for all, but needs to be proved for each particular theory, is essential. Indeed, if
 370 a theory is presented entirely by rewriting rules, completeness implies the consistency of the
 371 theory, since no rule can be applied on the empty set of clauses. Consequently, the proof of
 372 the completeness cannot be easier than the proof of consistency of the theory, and, according
 373 to Gödel, cannot be proven in the theory itself.

374 5.3 Beyond Deduction Modulo Theory

375 ► **Example 9.** Let us recall the set of clauses from the Introduction:

$$376 \frac{}{\neg X \in \mathcal{P}(Y) \vee \neg Z \in X \vee Z \in Y} \quad (1) \qquad \frac{}{X \in \mathcal{P}(Y) \vee d(X, Y) \in X} \quad (2)$$

$$377 \frac{}{X \in \mathcal{P}(Y) \vee \neg d(X, Y) \in Y} \quad (3)$$

378 Note that this example is not covered by Ordered Resolution with Selection, at least
 379 not if a simplification ordering is used, because we cannot have $X \in \mathcal{P}(Y) \succ \delta(X, Y) \in X$
 380 since with $\theta = \{X \mapsto \mathcal{P}(Z); Y \mapsto Z\}$ their instances are ordered in the wrong direction:
 381 $\mathcal{P}(Z) \in \mathcal{P}(Z) \prec \delta(\mathcal{P}(Z), Z) \in \mathcal{P}(Z)$.

382 The synthetic rules of the example from the Introduction correspond to the derivations
 383 when one of the clauses is focused. For instance, if we consider the clause (1), in a context
 384 Γ containing this clause, a proof putting the focus on $\ulcorner(1)\urcorner$ necessarily is of the following
 385 shape:

$$386 \frac{\frac{\frac{\widehat{\Downarrow} \vdash \Gamma \Downarrow \neg u \in \mathcal{P}(v) \vdash \quad \widehat{\Downarrow} \vdash \Gamma \Downarrow \neg t \in u \vdash}{\Downarrow \vee \vdash \Gamma \Downarrow \neg u \in \mathcal{P}(v) \vee \neg t \in u \vee t \in v \vdash} \quad \text{Store} \frac{\Gamma, t \in v \uparrow \vdash}{\Gamma \uparrow t \in v \vdash} \quad \text{Release} \frac{\Gamma \uparrow t \in v \vdash}{\Gamma \Downarrow t \in v \vdash}}{\Downarrow \vee \vdash \Gamma \Downarrow \forall X Y Z. \neg X \in \mathcal{P}(Y) \vee \neg Z \in X \vee Z \in Y \vdash} \quad \text{Focus} \frac{\Gamma \uparrow \vdash}{\Gamma \uparrow \vdash}}$$

387 where t, u, v are arbitrary terms, and where, to be able to close the left and middle branch,
 388 $u \in \mathcal{P}(v)$ and $t \in u$ must belong to Γ . So Γ is in fact of the form

389 $\forall X Y Z. \neg X \in \mathcal{P}(Y) \vee \neg Z \in X \vee Z \in Y, \Delta, u \in \mathcal{P}(v), t \in u$ for some Δ , and the axiom
 390 $\forall X Y Z. \neg X \in \mathcal{P}(Y) \vee \neg Z \in X \vee Z \in Y$ can be replaced by the synthetic rule:

$$391 (1) \frac{\Delta, u \in \mathcal{P}(v), t \in u, t \in v \uparrow \vdash}{\Delta, u \in \mathcal{P}(v), t \in u \uparrow \vdash} .$$

392 The computation of the other synthetic rules is left as an exercise for the reader.

393 The resolution rules given in the Introduction corresponds to the ground instances of
 394 Resolution with Selection with our three input clauses.

23:12 Linking Focusing and Resolution with Selection

395 The question that remains is how we can prove the completeness of such a selection. We can
396 in fact consider only subselections.

397 ► **Definition 10** (Singleton subselection). Given a selection function \mathcal{S} , the selection function
398 \mathcal{S}_1 is a singleton subselection of \mathcal{S} if

- 399 ■ $\mathcal{S}_1(C) \subseteq \mathcal{S}(C)$ for all C
- 400 ■ if $\mathcal{S}(C) \neq \emptyset$ then $\text{card}(\mathcal{S}_1(C)) = 1$

► **Example 11.** A singleton subselection of Example 9 can be

$$\neg X \in \mathcal{P}(Y) \vee \underline{\neg Z \in X} \vee Z \in Y \quad \underline{X \in \mathcal{P}(Y)} \vee d(X, Y) \in X \quad \underline{X \in \mathcal{P}(Y)} \vee \neg d(X, Y) \in Y$$

401 ► **Theorem 12.** *Resolution with input selection \mathcal{S} is complete iff for all singleton subselections*
402 *\mathcal{S}_1 of \mathcal{S} , Resolution with input selection \mathcal{S}_1 is complete.*

403 Since singleton subselections can be linked with rewriting systems in Deduction Modulo
404 Theory according to last subsection, we can reduce the problem of completeness in our
405 framework to several problems of completeness in Deduction Modulo Theory.

406 Conclusion and Further Work

407 We generalized focusing and resolution with selection, proved that they correspond, and
408 showed how known calculi are instances of this framework, namely ordinary focusing, hy-
409 perresolution, Deduction Modulo Theory and Superdeduction. We also showed how to
410 reduce completeness of this framework to several completeness proofs in Deduction Modulo
411 Theory. We can therefore reuse the various techniques for proving completeness in Deduction
412 Modulo Theory [25, 22, 19, 8] in our framework. As Deduction Modulo Theory already
413 gives significant results in industrial applications when the theory is a variant of set theory
414 (more precisely, set theory of the B method) [11], we can expect our framework to lead to
415 even better outcomes. The notable results presented here raise the following new areas of
416 investigations.

417 First, we need to study how to apply selection also in the generated clauses. This should
418 allow us to cover the cases of Ordered Resolution with Selection and of Semantic Resolution
419 in the first-order case. Dually, in the sequent calculus part, this would correspond to the
420 possibility to dynamically add selection in formulas of subderivations. This could probably
421 be linked with the work of Deplagne [17] where rewrite rules corresponding to induction
422 hypotheses are dynamically added in the rewriting system of a sequent calculus for Deduction
423 Modulo Theory. Note that we already have one direction, namely from Resolution with
424 Input Selection to LKF^\perp , since Lemmas 17, 19, and 20 do not assume anything on the
425 generated clauses; except, for **Factoring**, that it selects only instances of literals that were
426 already selected. The converse direction would require a meta-theorem of completeness, since
427 obviously it is not complete for all possible dynamic choices of selection.

428 Since focusing is defined not only for classical first-order logic but also for linear, intu-
429 itionistic, modal logics, the work in this paper could serve as a starting point to study how
430 to get automated proof search methods for these logics with a selection mechanism.

431 Another worthwhile point is how equality should be handled in our framework. In partic-
432 ular, it would be interesting to see how paramodulation calculi, in particular superposition,
433 can be embedded into a sequent calculus.

434 Finally, it would be worth investigating whether completeness proofs based on model
435 construction, such as semantic completeness proofs of tableaux (related to sequent calculus),
436 and completeness proof of superposition [2], can be related in our framework.

437 ——— **References** ———

- 438 **1** Jean-Marc Andreoli. Logic programming with focusing proofs in linear logic. *Journal of*
439 *Logic and Computation*, 2(3):297–347, 1992.
- 440 **2** L. Bachmair and H. Ganzinger. Rewrite-based equational theorem proving with selection
441 and simplification. *Journal of Logic and Computation*, 4(3):1–31, 1994.
- 442 **3** Leo Bachmair and Harald Ganzinger. Resolution theorem proving. In John Alan Robinson
443 and Andrei Voronkov, editors, *Handbook of Automated Reasoning*, pages 19–99. Elsevier
444 and MIT Press, 2001.
- 445 **4** Richard Bonichon and Olivier Hermant. A semantic completeness proof for TaMed. In
446 Miki Hermann and Andrei Voronkov, editors, *LPAR*, volume 4246 of *LNCS*, pages 167–
447 181. Springer, 2006.
- 448 **5** Paul Brauner, Clément Houtmann, and Claude Kirchner. Principle of superdeduction. In
449 Luke Ong, editor, *LICS*, pages 41–50, jul 2007.
- 450 **6** Guillaume Burel. Experimenting with deduction modulo. In Viorica Sofronie-Stokkermans
451 and Nikolaj Bjørner, editors, *CADE*, volume 6803 of *LNCS*, pages 162–176. Springer, 2011.
- 452 **7** Guillaume Burel. From axioms to rewriting rules. Available on author’s web page, 2013.
- 453 **8** Guillaume Burel. Cut admissibility by saturation. In Gilles Dowek, editor, *RTA-TLCA*,
454 volume 8560 of *LNCS*, pages 124–138. Springer, 2014.
- 455 **9** Guillaume Burel and Gilles Dowek. How can we prove that a proof search method is not an
456 instance of another? In *LFMTP*, ACM International Conference Proceeding Series, pages
457 84–87. ACM, 2009.
- 458 **10** Guillaume Burel and Claude Kirchner. Regaining cut admissibility in deduction modulo
459 using abstract completion. *Information and Computation*, 208(2):140–164, 2010.
- 460 **11** Guillaume Bury, David Delahaye, Damien Doligez, Pierre Halmagrand, and Olivier Her-
461 mant. Automated deduction in the B set theory using typed proof search and deduction
462 modulo. In Ansgar Fehnker, Annabelle McIver, Geoff Sutcliffe, and Andrei Voronkov, edi-
463 tors, *LPAR*, volume 35 of *EPiC Series in Computing*, pages 42–58. EasyChair, 2015. URL:
464 <http://www.easychair.org/publications/volume/LPAR-20>.
- 465 **12** Samuel R. Buss, editor. *Handbook of proof theory*. Studies in logic and the foundations of
466 mathematics. Elsevier, Amsterdam, 1998.
- 467 **13** Kaustuv Chaudhuri and Frank Pfenning. A focusing inverse method theorem prover for
468 first-order linear logic. In Robert Nieuwenhuis, editor, *CADE*, volume 3632 of *LNCS*, pages
469 69–83. Springer, 2005.
- 470 **14** Kaustuv Chaudhuri, Frank Pfenning, and Greg Price. A logical characterization of forward
471 and backward chaining in the inverse method. *Journal of Automated Reasoning*, 40(2-
472 3):133–177, 2008.
- 473 **15** Zakaria Chihani, Tomer Libal, and Giselle Reis. The proof certifier checkers. In
474 *TABLEAUX*, volume 9323 of *LNCS*, pages 201–210, Wroclaw, Poland, 2015. Springer.
- 475 **16** Zakaria Chihani, Dale Miller, and Fabien Renaud. Foundational proof certificates in first-
476 order logic. In Maria Paola Bonacina, editor, *CADE*, volume 7898 of *LNCS*, pages 162–177.
477 Springer, 2013.
- 478 **17** Eric Deplagne and Claude Kirchner. Induction as deduction modulo. Rapport
479 de recherche, LORIA, Nov 2004. URL: [http://www.loria.fr/publications/2004/
480 A04-R-468/A04-R-468.ps](http://www.loria.fr/publications/2004/A04-R-468/A04-R-468.ps).
- 481 **18** Gilles Dowek. What is a theory? In Helmut Alt and Afonso Ferreira, editors, *STACS*,
482 volume 2285 of *LNCS*, pages 50–64. Springer, 2002.
- 483 **19** Gilles Dowek. Truth values algebras and proof normalization. In Thorsten Altenkirch and
484 Conor McBride, editors, *TYPES*, volume 4502 of *LNCS*, pages 110–124. Springer, 2006.
- 485 **20** Gilles Dowek. Polarized resolution modulo. In Cristian S. Calude and Vladimiro Sassone,
486 editors, *IFIP TCS*, volume 323 of *IFIP AICT*, pages 182–196. Springer, 2010.

- 487 21 Gilles Dowek, Thérèse Hardin, and Claude Kirchner. Theorem proving modulo. *Journal*
488 *of Automated Reasoning*, 31(1):33–72, 2003.
- 489 22 Gilles Dowek and Benjamin Werner. Proof normalization modulo. *The Journal of Symbolic*
490 *Logic*, 68(4):1289–1316, 2003.
- 491 23 Gilles Dowek and Benjamin Werner. Arithmetic as a theory modulo. In Jürgen Giesl,
492 editor, *RTA*, volume 3467 of *LNCS*, pages 423–437. Springer, 2005.
- 493 24 Mahfuza Farooque, Stéphane Graham-Lengrand, and Assia Mahboubi. A bisimulation
494 between $DPLL(T)$ and a proof-search strategy for the focused sequent calculus. In Alberto
495 Momigliano, Brigitte Pientka, and Randy Pollack, editors, *LFMTP*, pages 3–14. ACM,
496 2013.
- 497 25 Olivier Hermant. *Méthodes Sémantiques en Dédution Modulo*. PhD thesis, École Poly-
498 technique, 2005.
- 499 26 Olivier Hermant. Resolution is cut-free. *Journal of Automated Reasoning*, 44(3):245–276,
500 2009.
- 501 27 Clément Houtmann. Axiom directed focusing. In Stephano Berardi, Ferruccio Damiani,
502 and Ugo de’Liguoro, editors, *Types for Proofs and Programs*, volume 5497 of *LNCS*, pages
503 169–185. Springer, 2008.
- 504 28 Chuck Liang and Dale Miller. Focusing and polarization in linear, intuitionistic, and classi-
505 cal logics. *Theoretical Computer Science*, 410(46):4747–4768, 2009. Abstract Interpretation
506 and Logic Programming: In honor of professor Giorgio Levi.
- 507 29 Sean McLaughlin and Frank Pfenning. Imogen: Focusing the polarized inverse method for
508 intuitionistic propositional logic. In Iliano Cervesato, Helmut Veith, and Andrei Voronkov,
509 editors, *LPAR*, volume 5330 of *Lecture Notes in Computer Science*, pages 174–181. Springer,
510 2008.
- 511 30 Dale Miller and Marco Volpe. Focused labeled proof systems for modal logic. In Martin
512 Davis, Ansgar Fehnker, Annabelle McIver, and Andrei Voronkov, editors, *LPAR*, volume
513 9450 of *LNCS*, pages 266–280. Springer, 2015.
- 514 31 J. A. Robinson. Automatic deduction with hyper-resolution. *International Journal of*
515 *Computer Mathematics*, 1:227–234, 1965.
- 516 32 Peter Schroeder-Heister. Cut elimination for logics with definitional reflection. In *Non-*
517 *classical Logics and Information Processing*, volume 619 of *LNCS*, pages 146–171. Springer,
518 1990.
- 519 33 James R. Slagle. Automatic theorem proving with renamable and semantic resolution. *J.*
520 *ACM*, 14(4):687–697, 1967.
- 521 34 Larry Wos, George A. Robinson, and Daniel F. Carson. Efficiency and completeness of the
522 set of support strategy in theorem proving. *J. ACM*, 12(4):536–541, 1965.

A Appendix

The two directions of the proof of Theorem 3 are given in the next sections.

A.1 From Focused Proofs to Resolution Derivations

We need a few lemmas to prove the first direction.

► **Lemma 13.** *For all sets of clauses Γ , for all clauses C_1, \dots, C_n and D such that $\mathcal{S}(C_i) = \emptyset$ for all i and $\mathcal{S}(D) = \emptyset$, if $\Gamma, C_1, \dots, C_n \rightsquigarrow^* \square$ and $\Gamma, D \rightsquigarrow^* \square$, then $\Gamma, C_1 \curlywedge D, \dots, C_n \curlywedge D \rightsquigarrow^* \square$ where $\mathcal{S}(C_i \curlywedge D) = \emptyset$ for all i .*

Proof. By induction on the derivation length of $\Gamma, C_1, \dots, C_n \rightsquigarrow^* \square$, generalizing on Γ .

The base case is when $\square \in \Gamma$. Then, trivially $\Gamma, C_1 \curlywedge D, \dots, C_n \curlywedge D \rightsquigarrow^* \square$.

For the inductive case, suppose that there exists C_{n+1} such that $\Gamma, C_1, \dots, C_n \rightsquigarrow C_{n+1}$ and $\Gamma, C_1, \dots, C_n, C_{n+1} \rightsquigarrow^* \square$.

There are two cases:

■ C_{n+1} is derived using other clauses than one of the C_i . We therefore have $\Gamma, C_1 \curlywedge D, \dots, C_n \curlywedge D \rightsquigarrow C_{n+1}$.

We can apply the induction hypothesis on $\Gamma, C_1, \dots, C_n, C_{n+1} \rightsquigarrow^* \square$, which can be viewed as $\Gamma, C_{n+1}, C_1, \dots, C_n \rightsquigarrow^* \square$, since by weakening we have $\Gamma, C_{n+1}, D \rightsquigarrow^* \square$. We obtain

$\Gamma, C_{n+1}, C_1 \curlywedge D, \dots, C_n \curlywedge D \rightsquigarrow^* \square$. By definition of \rightsquigarrow^* we therefore have $\Gamma, C_1 \curlywedge D, \dots, C_n \curlywedge D \rightsquigarrow^* \square$.

■ At least one of the parents of C_{n+1} is some C_i . Since no literal is selected in the C_i , they can only be side clauses of Resolution with Selection, or any clause in Resolution or Factoring. We can therefore derive $C_{n+1} \curlywedge D$ from $\Gamma, C_1 \curlywedge D, \dots, C_n \curlywedge D$ with the same inference rule that produced C_{n+1} . We can apply the induction hypothesis on $\Gamma, C_1, \dots, C_n, C_{n+1} \rightsquigarrow^* \square$ which gives us $\Gamma, C_1 \curlywedge D, \dots, C_n \curlywedge D, C_{n+1} \curlywedge D \rightsquigarrow^* \square$.

Hence $\Gamma, C_1 \curlywedge D, \dots, C_n \curlywedge D \rightsquigarrow^* \square$. ◀

► **Corollary 14.** *For all sets of clauses Γ , for all clauses C_1, \dots, C_n such that $\mathcal{S}(C_i) = \emptyset$ for all i . If $\Gamma, C_i \rightsquigarrow^* \square$ for all i then $\Gamma, C_1 \curlywedge \dots \curlywedge C_n \rightsquigarrow^* \square$ where $\mathcal{S}(C_1 \curlywedge \dots \curlywedge C_n) = \emptyset$.*

► **Lemma 15.** *For all sets of clauses Γ , for all substitutions θ , for all clauses C such that $\mathcal{S}(\theta C) = \emptyset$, if $\Gamma, \theta C \rightsquigarrow^* \square$ then $\Gamma, C \rightsquigarrow^* \square$ where $\mathcal{S}(C) = \emptyset$.*

Proof. By induction on the derivation length of $\Gamma, \theta C \rightsquigarrow^* \square$. As in the previous proof, the only interesting case is when the clause θC is used in the first step $\Gamma, \theta C \rightsquigarrow D$ of the derivation. If the first step $\Gamma, \theta C \rightsquigarrow D$ is Factoring, then $D = \sigma \theta C$ where σ is the most general unifier of two literals in θC . We can apply the induction hypothesis using $\sigma \theta$ instead of σ .

Otherwise, if the first step is Resolution with Selection or Resolution. Let θC be $L \curlywedge \theta C'$ where L is the literal of θC used in that step. Let C be $L_1 \curlywedge \dots \curlywedge L_n \curlywedge C'$ where the L_i are exactly the literals of C such that $\theta L_i = L$. The step produces therefore a clause $D = \sigma(\theta C' \curlywedge D')$ where σ is the most general unifier of a unification problem $L \stackrel{?}{=} L', Prob.$ Let $\omega = mgu(L_1, \dots, L_n)$, then $\theta = \theta' \omega$. First, from C one can derive $\omega L_1 \curlywedge \omega C'$ by repetitively applying Factoring to C . We have $\sigma L = \sigma L'$, hence $\sigma \theta' \omega L_1 = \sigma L'$. Since we rename variables in the premises of the resolution rules, we can assume that θ' does not affect

23:16 Linking Focusing and Resolution with Selection

566 the variables of L' . Consequently, $\sigma\theta'\omega L_1 = \sigma\theta'L'$. Thus, $\sigma\theta$ is a solution of the unification
 567 problem $\omega L_1 =? L', Prob$. Let μ be its most general solution. Therefore, there exists κ such
 568 that $\sigma\theta' = \kappa\mu$.

569 From $\omega L_1 \gamma \omega C'$ and the same other clauses as in the step $\Gamma, \theta C \rightsquigarrow D$, we can therefore
 570 derive $\mu(\omega C' \gamma D')$. Once again, the variables of D' , coming from the other clauses, can
 571 be assumed to be distinct of those of C , therefore $D' = \theta'D'$. We have $\sigma(\theta C' \gamma D') =$
 572 $\sigma(\theta'\omega C' \gamma D') = \sigma\theta'(\omega C' \gamma D') = \kappa\mu(\omega C' \gamma D)$. We can therefore apply the induction
 573 hypothesis, using κ on $\mu(\omega C' \gamma D)$ instead of θ on C . \blacktriangleleft

574 **► Theorem 16.** *If $\ulcorner \Gamma \urcorner \uparrow \vdash$, then $\Gamma \rightsquigarrow^* \square$.*

575 **Proof.** By induction on the proof $\ulcorner \Gamma \urcorner \uparrow \vdash$. We generalize the statement a little by allowing
 576 the sequent to contain not only translation of clauses but also literals with a positive polarity.
 577 On such a sequent, only two rules can be applied, namely $\widehat{\uparrow} \vdash$ and **Focus**. Since $\ulcorner \Gamma \urcorner$ contains
 578 only formulas in clausal form, there are only four cases:

579 $\blacksquare \frac{\widehat{\uparrow} \vdash \ulcorner \Gamma' \urcorner, L, L^\perp \uparrow \vdash}{\ulcorner \Gamma' \urcorner, L, L^\perp \uparrow \vdash}$

580 In that case, we can simply apply Resolution on L and L^\perp to derive \square , hence $\Gamma', L, L^\perp \rightsquigarrow \square$.

581 \blacksquare The proof focuses on a formula corresponding to the empty clause:

582 **Focus** $\frac{\downarrow \perp \vdash \ulcorner \Gamma \urcorner \downarrow \perp \vdash}{\ulcorner \Gamma \urcorner \uparrow \vdash}$

583 In that case, \square already belongs to Γ .

584 \blacksquare The proof focuses on a formula corresponding to a clause without selection. Because of
 585 focusing constraints, the proof is necessarily of the form:

586 **Store** $\frac{\ulcorner \Gamma \urcorner, \theta L_1 \uparrow \vdash}{\ulcorner \Gamma \urcorner \uparrow \theta L_1 \vdash} \quad \dots \quad \text{Store} \frac{\ulcorner \Gamma \urcorner, \theta L_m \uparrow \vdash}{\ulcorner \Gamma \urcorner \uparrow \theta L_m \vdash}$
 $\uparrow \vee \vdash \frac{\ulcorner \Gamma \urcorner \uparrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash}{\ulcorner \Gamma \urcorner \uparrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash}$
Release $\frac{\ulcorner \Gamma \urcorner \uparrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash}{\ulcorner \Gamma \urcorner \downarrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash}$
 $\downarrow \vee \vdash \frac{\ulcorner \Gamma \urcorner \downarrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash}{\ulcorner \Gamma \urcorner \downarrow \overline{\forall x}. \delta^-(L_1 \vee^+ \dots \vee^+ L_m) \vdash}$
Focus $\frac{\ulcorner \Gamma \urcorner \downarrow \overline{\forall x}. \delta^-(L_1 \vee^+ \dots \vee^+ L_m) \vdash}{\ulcorner \Gamma \urcorner \uparrow \vdash}$

587 By induction hypothesis, we have derivations of $\Gamma, \theta L_k \rightsquigarrow^* \square$ for all $1 \leq k \leq m$. By
 588 Corollary 14, we have a derivation $\Gamma, \theta(L_1 \gamma \dots \gamma L_k) \rightsquigarrow^* \square$ with nothing selected in
 589 $\theta(L_1 \gamma \dots \gamma L_k)$. By Lemma 15, we have a derivation of $\Gamma \rightsquigarrow^* \square$, with $L_1 \gamma \dots \gamma L_k$ in
 590 Γ .

591 \blacksquare The proof focuses on a formula corresponding to a clause with selection. Because of
 592 focusing constraints, the proof is necessarily of the form:

593 $\dots \widehat{\downarrow} \vdash \frac{\ulcorner \Gamma \urcorner \downarrow \theta K_j \vdash}{\ulcorner \Gamma \urcorner \downarrow \theta K_j \vdash} \quad \dots \quad \uparrow \vee \vdash \frac{\dots \text{Store} \frac{\ulcorner \Gamma \urcorner, \theta L_k \uparrow \vdash}{\ulcorner \Gamma \urcorner \uparrow \theta L_k \vdash} \quad \dots}{\ulcorner \Gamma \urcorner \uparrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash}$
Release $\frac{\ulcorner \Gamma \urcorner \uparrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash}{\ulcorner \Gamma \urcorner \downarrow \delta^-(L_1 \vee^+ \dots \vee^+ L_m) \vdash}$
 $\downarrow \vee \vdash \frac{\ulcorner \Gamma \urcorner \downarrow \delta^-(L_1 \vee^+ \dots \vee^+ L_m) \vdash}{\ulcorner \Gamma \urcorner \downarrow \theta(\underline{K}_1 \vee^- \dots \vee^- \underline{K}_n \vee^- \delta^-(L_1 \vee^+ \dots \vee^+ L_m)) \vdash}$
 $\downarrow \vee \vdash \frac{\ulcorner \Gamma \urcorner \downarrow \theta(\underline{K}_1 \vee^- \dots \vee^- \underline{K}_n \vee^- \delta^-(L_1 \vee^+ \dots \vee^+ L_m)) \vdash}{\ulcorner \Gamma \urcorner \downarrow \overline{\forall x}. \underline{K}_1 \vee^- \dots \vee^- \underline{K}_n \vee^- \delta^-(L_1 \vee^+ \dots \vee^+ L_m) \vdash}$
Focus $\frac{\ulcorner \Gamma \urcorner \downarrow \overline{\forall x}. \underline{K}_1 \vee^- \dots \vee^- \underline{K}_n \vee^- \delta^-(L_1 \vee^+ \dots \vee^+ L_m) \vdash}{\ulcorner \Gamma \urcorner \uparrow \vdash}$

594 where $\overline{\forall x}. \underline{K}_1 \vee^- \dots \vee^- \underline{K}_n \vee^- (L_1 \vee^+ \dots \vee^+ L_m)$ and θK_j^\perp for all $1 \leq j \leq n$ are members
 595 of $\ulcorner \Gamma \urcorner$.

596 By induction hypothesis, we have derivations of $\Gamma, \theta L_k \rightsquigarrow^* \square$ for all $1 \leq k \leq m$. By
 597 Corollary 14, we have a derivation $\Gamma, \theta(L_1 \gamma \dots \gamma L_k) \rightsquigarrow^* \square$.

598 From $\underline{K}_1 \curlywedge \dots \curlywedge \underline{K}_n \curlywedge L_1 \curlywedge \dots \curlywedge L_m$, since the application of $\widehat{\Downarrow} \vdash$ above imposes that all
 599 θK_j^\perp are in Γ , we can apply Resolution with Selection to obtain $\theta(L_1 \curlywedge \dots \curlywedge L_m)$, hence
 600 a derivation $\Gamma \rightsquigarrow^* \square$. \blacktriangleleft

601 A.2 From Resolution Derivations to Focused Proofs

602 We prove that all inference rules of Resolution with Input Selection are admissible in LKF^\perp :
 603 if $\Gamma \rightsquigarrow C$ then LKF^\perp proofs of $\ulcorner \Gamma \urcorner, \ulcorner C \urcorner \vdash$ can be turned into proofs of $\ulcorner \Gamma \urcorner \vdash$.

604 **► Lemma 17.** *For all sets of formulas Γ , for all clauses C , for all substitutions σ , assuming*
 605 $\sigma\mathcal{S}(C) \subseteq \mathcal{S}(\sigma C)$,
 606 *if $\Gamma, \ulcorner C \urcorner, \ulcorner \sigma C \urcorner \uparrow \vdash$, then $\Gamma, \ulcorner C \urcorner \uparrow \vdash$.*

607 **Proof.** By induction on the proof $\Gamma, \ulcorner C \urcorner, \ulcorner \sigma C \urcorner \uparrow \vdash$. If the proof does not begin by focusing
 608 on $\ulcorner \sigma C \urcorner$, this is a simple application of the induction hypothesis (considering coarse grain
 609 proof steps consisting of an alternation of a synchronous and an asynchronous phases).
 610 Otherwise, let σC be $\underline{K}_1 \curlywedge \dots \curlywedge \underline{K}_n \curlywedge L_1 \curlywedge \dots \curlywedge L_m$. The proof begins with

$$\begin{array}{c}
 \dots \widehat{\Downarrow} \vdash \frac{\Gamma' \Downarrow \theta \underline{K}_j \vdash \dots}{\Gamma' \Downarrow \theta(\underline{K}_1 \vee^- \dots \vee^- \underline{K}_n \vee^- (L_1 \vee^+ \dots \vee^+ L_m)) \vdash} \dots \\
 \Downarrow \vee \vdash \frac{\dots \text{Store } \frac{\Gamma', \theta L_k \uparrow \vdash}{\Gamma' \uparrow \theta L_k \vdash} \dots}{\Gamma' \uparrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash} \dots \\
 \text{Release } \frac{\dots \text{Store } \frac{\Gamma', \theta L_k \uparrow \vdash}{\Gamma' \uparrow \theta L_k \vdash} \dots}{\Gamma' \uparrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash} \dots \\
 \Downarrow \vee \vdash \frac{\dots \text{Store } \frac{\Gamma', \theta L_k \uparrow \vdash}{\Gamma' \uparrow \theta L_k \vdash} \dots}{\Gamma' \uparrow \theta(L_1 \vee^+ \dots \vee^+ L_m) \vdash} \dots \\
 \text{Focus } \frac{\Gamma' \Downarrow \theta(\underline{K}_1 \vee^- \dots \vee^- \underline{K}_n \vee^- (L_1 \vee^+ \dots \vee^+ L_m)) \vdash}{\Gamma' \Downarrow \overline{\forall x}. \underline{K}_1 \vee^- \dots \vee^- \underline{K}_n \vee^- (L_1 \vee^+ \dots \vee^+ L_m) \vdash} \Gamma' \uparrow \vdash
 \end{array}$$

where $\Gamma' = \Gamma, \ulcorner C \urcorner, \ulcorner \sigma C \urcorner$ and

612 for all $1 \leq j \leq n$ the literal θK_j^\perp is in Γ .

613 Let C be

614 $\underline{K}_1^1 \curlywedge \dots \curlywedge \underline{K}_1^{k_1} \curlywedge \dots \curlywedge \underline{K}_n^1 \curlywedge \dots \curlywedge \underline{K}_n^{k_n} \curlywedge L_1^1 \curlywedge \dots \curlywedge L_1^{l_1} \curlywedge \dots \curlywedge L_m^1 \curlywedge \dots \curlywedge L_m^{l_m}$ where $\sigma K_i^j = K_i$ and $\sigma L_i^j =$

615 L_i for all i, j . By hypothesis, the literals selected in C are among the \widehat{K}_i^j . For all i, j , one can

616 either build a proof $\widehat{\Downarrow} \vdash \frac{\Gamma, \ulcorner C \urcorner \Downarrow \theta \sigma K_i^j \vdash}{\Gamma, \ulcorner C \urcorner \Downarrow \theta \sigma K_i^j \vdash}$ if it is selected, or $\text{Store } \frac{\Gamma, \ulcorner C \urcorner, \theta \sigma K_i^j \uparrow \vdash}{\Gamma, \ulcorner C \urcorner \uparrow \theta \sigma K_i^j \vdash}$

617 if it is not.

618 We apply the induction hypothesis on $\Gamma', \theta L_k \uparrow \vdash$, hence we have proofs $\Gamma, \ulcorner C \urcorner, \theta L_k \uparrow \vdash$
 619 for all k .

620 We can therefore build the proof

$$\begin{array}{c}
 \dots \widehat{\Downarrow} \vdash \frac{\Gamma, \ulcorner C \urcorner \Downarrow \theta \sigma K_i^j \vdash \dots}{\Gamma, \ulcorner C \urcorner \Downarrow \theta \sigma K_i^j \vdash} \dots \\
 \Downarrow \vee \vdash \frac{\dots \text{Store } \frac{\Gamma, \ulcorner C \urcorner, \theta \sigma K_i^j \uparrow \vdash}{\Gamma, \ulcorner C \urcorner \uparrow \theta \sigma K_i^j \vdash} \dots \quad \dots \quad \Gamma, \ulcorner C \urcorner, \theta L_i \uparrow \vdash}{\Gamma, \ulcorner C \urcorner \uparrow \dots \vee^+ \theta \sigma K_i^m \vee^+ \dots \vee^+ \theta \sigma L_i^j \vee^+ \dots \vdash} \dots \\
 \text{Release } \frac{\dots \text{Store } \frac{\Gamma, \ulcorner C \urcorner, \theta \sigma K_i^j \uparrow \vdash}{\Gamma, \ulcorner C \urcorner \uparrow \theta \sigma K_i^j \vdash} \dots \quad \dots \quad \Gamma, \ulcorner C \urcorner, \theta L_i \uparrow \vdash}{\Gamma, \ulcorner C \urcorner \uparrow \dots \vee^+ \theta \sigma K_i^m \vee^+ \dots \vee^+ \theta \sigma L_i^j \vee^+ \dots \vdash} \dots \\
 \Downarrow \vee \vdash \frac{\dots \text{Store } \frac{\Gamma, \ulcorner C \urcorner, \theta \sigma K_i^j \uparrow \vdash}{\Gamma, \ulcorner C \urcorner \uparrow \theta \sigma K_i^j \vdash} \dots \quad \dots \quad \Gamma, \ulcorner C \urcorner, \theta L_i \uparrow \vdash}{\Gamma, \ulcorner C \urcorner \uparrow \dots \vee^+ \theta \sigma K_i^m \vee^+ \dots \vee^+ \theta \sigma L_i^j \vee^+ \dots \vdash} \dots \\
 \text{Focus } \frac{\Gamma, \ulcorner C \urcorner \Downarrow \dots \vee^- \theta \sigma K_i^j \vee^- \dots \vee^- (\dots \vee^+ \theta \sigma K_i^m \vee^+ \dots \vee^+ \theta \sigma L_i^j \vee^+ \dots) \vdash}{\Gamma, \ulcorner C \urcorner \Downarrow \overline{\forall x}. \dots \vee^- \underline{K}_i^j \vee^- \dots \vee^- (\dots \vee^+ K_i^m \vee^+ \dots \vee^+ L_i^j \vee^+ \dots) \vdash} \Gamma, \ulcorner C \urcorner \uparrow \vdash
 \end{array}$$

\blacktriangleleft

622 **► Corollary 18.** *Factoring is admissible in LKF^\perp .*

623 **► Lemma 19.** *Resolution is admissible in LKF^\perp :*

624 *For all set of formulas Γ , for all clauses $L \curlywedge C$ and $L'^\perp \curlywedge D$ without selection, if $\sigma =$*
 625

626 $mgu(L, L')$,
 627 if $\Gamma, \lceil L \vee C \rceil, \lceil L'^\perp \vee D \rceil, \lceil \sigma(C \vee D) \rceil \uparrow \vdash$
 628 then $\Gamma, \lceil L \vee C \rceil, \lceil L'^\perp \vee D \rceil \uparrow \vdash$.

629 **Proof.** By induction on the proof
 630 $\Gamma, \lceil L \vee C \rceil, \lceil L'^\perp \vee D \rceil, \lceil \sigma(C \vee D) \rceil \uparrow \vdash$.

631 Let Γ' be $\Gamma, \lceil L \vee C \rceil, \lceil L'^\perp \vee D \rceil$ and Γ'' be $\Gamma', \lceil \sigma(C \vee D) \rceil$. If the proof does not begin by
 632 focusing on $\lceil \sigma(C \vee D) \rceil$, this is a simple application of the induction hypothesis. Otherwise,
 633 let $\sigma(C \vee D)$ be $\underline{I}_1 \vee \dots \vee \underline{I}_n \vee J_1 \vee \dots \vee J_m$. The proof begins with

$$\begin{array}{c}
 \dots \quad \widehat{\Downarrow} \vdash \frac{\Gamma'' \Downarrow \theta \underline{I}_j \vdash \dots}{\Gamma'' \Downarrow \theta (J_1 \vee^+ \dots \vee^+ J_m) \vdash} \quad \dots \quad \text{Release} \quad \frac{\uparrow \vee \vdash \frac{\dots \text{Store} \frac{\Gamma'', \theta J_k \uparrow \vdash}{\Gamma'' \uparrow \theta J_k \vdash} \dots}{\Gamma'' \uparrow \theta (J_1 \vee^+ \dots \vee^+ J_m) \vdash}}{\Gamma'' \Downarrow \theta (J_1 \vee^+ \dots \vee^+ J_m) \vdash} \\
 \Downarrow \vee \vdash \frac{\dots}{\Gamma'' \Downarrow \theta (\underline{I}_1 \vee^- \dots \vee^- \underline{I}_n \vee^- (J_1 \vee^+ \dots \vee^+ J_m)) \vdash} \quad \dots \quad \text{Focus} \quad \frac{\Downarrow \vee \vdash \frac{\Gamma'' \Downarrow \overline{\forall x}. \underline{I}_1 \vee^- \dots \vee^- \underline{I}_n \vee^- (J_1 \vee^+ \dots \vee^+ J_m) \vdash}{\Gamma'' \uparrow \vdash}}{\Gamma'' \uparrow \vdash}
 \end{array}$$

where, to be able to close

635 the left branches, for all $1 \leq j \leq n$ the literal θI_j^\perp is in Γ' .

636 We know that C is $\dots \vee I_i^1 \vee \dots \vee I_i^{k_i} \vee \dots \vee J_j^1 \vee \dots \vee J_j^{l_j} \vee \dots$ where i ranges over a subset
 637 of $\{1, \dots, n\}$ and j over a subset of $\{1, \dots, m\}$, and $\sigma I_i^x = I_i$ and $\sigma J_j^y = J_j$ for all x, y .
 638 Likewise, D is $\dots \vee I_i^1 \vee \dots \vee I_i^{k_i} \vee \dots \vee J_j^1 \vee \dots \vee J_j^{l_j} \vee \dots$ where i ranges over a subset of
 639 $\{1, \dots, n\}$ and j over a subset of $\{1, \dots, m\}$, and $\sigma I_i^x = I_i$ and $\sigma J_j^y = J_j$ for all x, y .

640 We apply the induction hypothesis on $\Gamma'', \theta J_k \uparrow \vdash$, hence we have proof $\Gamma, \lceil L \vee C \rceil, \lceil L'^\perp \vee D \rceil, \theta J_k \uparrow \vdash$.

641 We can build the following proof of $\Gamma' \uparrow \vdash$: first, we focus on $\lceil L \vee C \rceil$ to get the derivation

$$\begin{array}{c}
 \text{Store} \frac{\Gamma', \theta \sigma L \uparrow \vdash}{\Gamma' \uparrow \theta \sigma L \vdash} \quad \dots \quad \text{Store} \frac{\Gamma', \theta I_i \uparrow \vdash}{\Gamma' \uparrow \theta I_i \vdash} \quad \dots \quad \text{Store} \frac{\Gamma', \theta J_j \uparrow \vdash}{\Gamma' \uparrow \theta J_j \vdash} \quad \dots \\
 \uparrow \vee \vdash \frac{\dots}{\Gamma' \uparrow \theta \sigma (L \vee^+ \dots \vee^+ I_i^1 \vee^+ \dots \vee^+ I_i^{k_i} \vee^+ \dots \vee^+ J_j^1 \vee^+ \dots \vee^+ J_j^{l_j} \vee^+ \dots) \vdash} \\
 \text{Release} \frac{\Gamma' \uparrow \theta \sigma (L \vee^+ \dots \vee^+ I_i^1 \vee^+ \dots \vee^+ I_i^{k_i} \vee^+ \dots \vee^+ J_j^1 \vee^+ \dots \vee^+ J_j^{l_j} \vee^+ \dots) \vdash}{\Gamma' \Downarrow \theta \sigma (L \vee^+ \dots \vee^+ I_i^1 \vee^+ \dots \vee^+ I_i^{k_i} \vee^+ \dots \vee^+ J_j^1 \vee^+ \dots \vee^+ J_j^{l_j} \vee^+ \dots) \vdash} \\
 \Downarrow \vee \vdash \frac{\dots}{\Gamma' \Downarrow \overline{\forall x}. L \vee^+ \dots \vee^+ I_i^1 \vee^+ \dots \vee^+ I_i^{k_i} \vee^+ \dots \vee^+ J_j^1 \vee^+ \dots \vee^+ J_j^{l_j} \vee^+ \dots \vdash} \\
 \text{Focus} \frac{\dots}{\Gamma' \uparrow \vdash}
 \end{array}$$

643 In this derivation, all right branches are closed by induction hypothesis. On the left branch,
 644 we focus on $\lceil L'^\perp \vee D \rceil$. Let Γ''' be $\Gamma', \theta \sigma L$. We get the derivation

$$\begin{array}{c}
 \uparrow \vdash \frac{\Gamma''', \theta \sigma L'^\perp \uparrow \vdash}{\Gamma''' \uparrow \theta \sigma L'^\perp \vdash} \quad \dots \quad \uparrow \vdash \frac{\Gamma''', \theta I_i \uparrow \vdash}{\Gamma''' \uparrow \theta I_i \vdash} \quad \dots \quad \text{Store} \frac{\Gamma''', \theta J_j \uparrow \vdash}{\Gamma''' \uparrow \theta J_j \vdash} \quad \dots \\
 \uparrow \vee \vdash \frac{\dots}{\Gamma''' \uparrow \theta \sigma (L'^\perp \vee^+ \dots \vee^+ I_i^1 \vee^+ \dots \vee^+ I_i^{k_i} \vee^+ \dots \vee^+ J_j^1 \vee^+ \dots \vee^+ J_j^{l_j} \vee^+ \dots) \vdash} \\
 \text{Release} \frac{\Gamma''' \uparrow \theta \sigma (L'^\perp \vee^+ \dots \vee^+ I_i^1 \vee^+ \dots \vee^+ I_i^{k_i} \vee^+ \dots \vee^+ J_j^1 \vee^+ \dots \vee^+ J_j^{l_j} \vee^+ \dots) \vdash}{\Gamma''' \Downarrow \theta \sigma (L'^\perp \vee^+ \dots \vee^+ I_i^1 \vee^+ \dots \vee^+ I_i^{k_i} \vee^+ \dots \vee^+ J_j^1 \vee^+ \dots \vee^+ J_j^{l_j} \vee^+ \dots) \vdash} \\
 \Downarrow \vee \vdash \frac{\dots}{\Gamma''' \Downarrow \overline{\forall x}. L'^\perp \vee^+ \dots \vee^+ I_i^1 \vee^+ \dots \vee^+ I_i^{k_i} \vee^+ \dots \vee^+ J_j^1 \vee^+ \dots \vee^+ J_j^{l_j} \vee^+ \dots \vdash} \\
 \text{Focus} \frac{\dots}{\Gamma''' \uparrow \vdash}
 \end{array}$$

646 The left branch can be closed because $\theta \sigma L = \theta \sigma L'$. The right branches are closed by
 647 induction hypothesis. \blacktriangleleft

648 **► Lemma 20.** Resolution with Selection is admissible in LKF^\perp :

649 For all set of formulas Γ , for all clauses $\underline{K}_1 \vee \dots \vee \underline{K}_n \vee C$, $K_1'^\perp \vee D_1, \dots$, and $K_n'^\perp \vee D_n$,
 650 where $\mathcal{S}(\underline{K}_1 \vee \dots \vee \underline{K}_n \vee C) = \{K_1; \dots; K_n\}$, $\mathcal{S}(K_i'^\perp \vee D_i) = \emptyset$ and σ is the most general
 651 unifier of the simultaneous unification problem $K_1 =? K_1', \dots, K_n =? K_n'$,

652 if $\Gamma, \ulcorner \underline{K}_1 \urcorner \dots \urcorner \underline{K}_n \urcorner C^\urcorner, \dots, \ulcorner K'_i \urcorner \urcorner D_i \urcorner, \dots, \ulcorner \sigma(C \urcorner D_1 \urcorner \dots \urcorner D_n) \urcorner \uparrow \vdash$ then $\Gamma, \ulcorner \underline{K}_1 \urcorner \dots \urcorner \underline{K}_n \urcorner C^\urcorner, \dots, \ulcorner K'_i \urcorner \urcorner D_i \urcorner, \dots \uparrow \vdash$.

653 **Proof.** By induction on the proof

654 $\Gamma, \ulcorner \underline{K}_1 \urcorner \dots \urcorner \underline{K}_n \urcorner C^\urcorner, \dots, \ulcorner K'_i \urcorner \urcorner D_i \urcorner, \dots, \ulcorner \sigma(C \urcorner D_1 \urcorner \dots \urcorner D_n) \urcorner \uparrow \vdash$. We follow the same
655 idea as in the proofs of the two precedent lemmas. If the proof does not begin by focusing on
656 $\ulcorner \sigma(C \urcorner D_1 \urcorner \dots \urcorner D_n) \urcorner$, this is a simple application of the induction hypothesis. Otherwise,
657 let Γ' be

658 $\Gamma, \ulcorner \underline{K}_1 \urcorner \dots \urcorner \underline{K}_n \urcorner C^\urcorner, \dots, \ulcorner K'_i \urcorner \urcorner D_i \urcorner, \dots$ and Γ'' be

659 $\Gamma', \ulcorner \sigma(C \urcorner D_1 \urcorner \dots \urcorner D_n) \urcorner$. Focusing leads us either to sequents $\Gamma'' \Downarrow \theta I_j \vdash$, with $\theta I_j \perp$ in
660 Γ , or to sequents $\Gamma'', \theta J_k \uparrow \vdash$ upon which one can apply the induction hypothesis. Let us
661 remark that for each literal L of C or D_i , $\theta \sigma L$ is either one of θI_j or one of θJ_k . Therefore,
662 we know how to close proofs of $\Gamma' \uparrow \theta \sigma L \vdash$ for each, either by induction hypothesis or using
663 $\theta I_j \perp$ in Γ .

664 To build the proof of $\Gamma' \uparrow \vdash$, we first focus on $\ulcorner K'_1 \urcorner \urcorner D_1 \urcorner$, instantiating the variables
665 using the substitution $\theta \sigma$. As explain above, we know how to close the branches coming
666 from D_1 , it remains the branch $\Gamma', \theta \sigma K'_1 \perp \uparrow \vdash$.

667 We do the same, focusing on $\ulcorner K'_2 \urcorner \urcorner D_2 \urcorner$ then ... then $\ulcorner K'_n \urcorner \urcorner D_n \urcorner$ and the remaining
668 branch is $\Gamma', \theta \sigma K'_1 \perp, \dots, \theta \sigma K'_n \perp \uparrow \vdash$.

669 We can close the proof by focusing on $\ulcorner \underline{K}_1 \urcorner \dots \urcorner \underline{K}_n \urcorner C^\urcorner$. Branches coming from C
670 can be closed as before, and the other branches are closed by

671 $\Downarrow \vdash \frac{\Gamma', \theta \sigma K'_1 \perp, \dots, \theta \sigma K'_n \perp \Downarrow \theta \sigma K_i \vdash}{\Gamma', \theta \sigma K'_1 \perp, \dots, \theta \sigma K'_n \perp \Downarrow \theta \sigma K_i \vdash}$ since $\theta \sigma K'_i = \theta \sigma K_i$ for all i . ◀

672 ▶ **Theorem 21.** For all set of clauses Γ , if $\Gamma \rightsquigarrow^* \square$, then $\ulcorner \Gamma \urcorner \uparrow \vdash$.

673 **Proof.** By induction on the length of the derivation $\Gamma \rightsquigarrow^* \square$. If \square is in Γ , then we focus on
674 $\ulcorner \square \urcorner = \perp$ and apply $\Downarrow \perp \vdash$. If the first step is **Factoring**, we apply Lemma 17. If it is **Resolution**,
675 we apply Lemma 19. If it is **Resolution with Selection**, we apply Lemma 20. ◀

676 A.3 Proofs of Section 5

677 **Proof of Theorem 8.** Proofs in both calculi correspond almost exactly. $\top \vdash$, $\wedge \vdash$, $\vee \vdash$ and $\exists \vdash$
678 in PUSC^\perp correspond exactly to $\uparrow \top \vdash$, $\uparrow \wedge \vdash$, $\uparrow \vee \vdash$ and $\uparrow \exists \vdash$ in LKF^\perp , except that if the top
679 connective of the subformulas in the premise(s) is \vee or \perp , or if they are literals, they have to
680 be put on the left hand side of \uparrow using **Store**. The translation of $\perp \vdash$ corresponds to $\Downarrow \perp \vdash$,
681 except that the latter can only be applied when there is no formula with positive polarity:

682 $\perp \vdash \frac{N_1, \dots, \perp, \dots, N_n \vdash}{N_1, \dots, \perp, \dots, N_n \vdash}$ becomes **Focus** $\frac{\Downarrow \perp \vdash \frac{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, \perp, \dots, |N_n| \Downarrow \perp \vdash}{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, \perp, \dots, |N_n| \uparrow \vdash}}{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, \perp, \dots, |N_n| \uparrow \vdash}$.

683 Similarly, $\vee \vdash$ corresponds to $\Downarrow \vee \vdash$, with the same proviso that there are no formulas with
684 positive polarity:

685 $\vee \vdash \frac{N_1, \dots, N_n, \forall x. A, \{t/x\}A \vdash}{N_1, \dots, N_n, \forall x. A \vdash}$ becomes

686 **Release** $\frac{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, |N_n|, \forall x. \delta^- |A| \uparrow \{t/x\}A \vdash}{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, |N_n|, \forall x. \delta^- |A| \Downarrow \delta^- \{t/x\}A \vdash}$
 $\Downarrow \vee \vdash \frac{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, |N_n|, \forall x. \delta^- |A| \Downarrow \delta^- \{t/x\}A \vdash}{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, |N_n|, \forall x. \delta^- |A| \Downarrow \forall x. \delta^- |A| \vdash}$
Focus $\frac{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, |N_n|, \forall x. \delta^- |A| \Downarrow \forall x. \delta^- |A| \vdash}{\llbracket \mathcal{R} \rrbracket, |N_1|, \dots, |N_n|, \forall x. \delta^- |A| \uparrow \vdash}$

687 with an extra **Store** step if the top connective of $\{t/x\}A$ is \vee or \perp or if it is a literal.

688 For the unfolding rules, if P rewrites positively to A , then there exists a rule $Q \rightarrow^+ B$
689 and a substitution θ such that $P = \theta Q$ and $A = \theta B$. This rule corresponds to a formula
690 $\llbracket Q \rightarrow^+ B \rrbracket = \overline{\forall x. Q \vee^- \delta^+ |B^\perp|}$. Always with the proviso that there is no formula with positive
691 polarity, let $\Gamma = \llbracket \mathcal{R}' \rrbracket, \overline{\forall x. Q \vee^- \delta^+ |B^\perp|}, |N_1|, \dots, |N_n|, \neg P$, then $\uparrow^+ \vdash$ therefore corresponds

692 to

$$\begin{array}{c}
 \widehat{\Downarrow} \vdash \frac{\Gamma \Downarrow \underline{P} \vdash}{\Downarrow \forall \vdash} \quad \text{Release} \frac{\Gamma \uparrow |A^\perp| \vdash}{\Gamma \Downarrow \delta^+ |A^\perp| \vdash} \\
 \hline
 \Downarrow \forall \vdash \frac{\Gamma \Downarrow \underline{\theta Q} \vee^- \delta^+ |\theta B^\perp| \vdash}{\Gamma \Downarrow \forall x. \underline{Q} \vee^- \delta^+ |B^\perp| \vdash} \\
 \hline
 \text{Focus} \frac{\Gamma \uparrow \vdash}{\Gamma \uparrow \vdash}
 \end{array}$$

694 with an extra **Store** step if the top connective of $|A^\perp|$ is \forall or \perp , or if it is a literal. Reciprocally,
 695 given a sequent $\llbracket \mathcal{R}' \rrbracket, \forall x. \underline{Q} \vee^- \delta^+ |B^\perp|, \Gamma \uparrow \vdash$, if we apply a **Focus** on $\forall x. \underline{Q} \vee^- \delta^+ |B^\perp|$, the
 696 derivation is necessarily of the same shape as above, so that there must be a literal $-\theta Q$ in
 697 Γ . The derivation therefore corresponds to an unfolding of $-\theta Q$ into $|\theta B^\perp|$.

698 The case of a negative rewriting is dual.

699 There remains to be proved that, in PUSC^\perp , the rules $\perp \vdash, \forall \vdash$ and $\uparrow \vdash$ can be delayed
 700 until the other rules are no longer applicable. This can be done by showing that these rules
 701 permute with the other ones. Note that this fact can be related with the strategy used in
 702 Tamed [4], a tableaux method based on Deduction Modulo Theory, where rules for universal
 703 quantifiers and for rewriting are applied when no other rules can be. ◀

704 **Proof sketch of Theorem 12.** Showing that LKF^\perp is complete amounts to proving that the
 705 cut rule

$$\text{cut} \frac{\Gamma, A \uparrow \vdash \quad \Gamma, A^\perp \uparrow \vdash}{\Gamma \uparrow \vdash}$$

707 is admissible.

708 Using the same techniques as in [10], we can try to eliminate cuts using structural cut
 709 elimination à la Pfenning. The only problematic case is when we cut around a literal that is
 710 used in focused instance in both branches:

$$\begin{array}{c}
 \widehat{\Downarrow} \vdash \frac{\Gamma', A \Downarrow \underline{A^\perp} \vdash}{\vdots} \quad \widehat{\Downarrow} \vdash \frac{\Gamma'', A^\perp \Downarrow \underline{A} \vdash}{\vdots} \\
 \hline
 \text{Focus} \frac{\Gamma, A \Downarrow E[\underline{A^\perp}] \vdash}{\text{cut} \frac{\Gamma, A \uparrow \vdash}{\Gamma \uparrow \vdash}} \quad \text{Focus} \frac{\Gamma, A^\perp \Downarrow F[\underline{A}] \vdash}{\Gamma, A^\perp \uparrow \vdash} \\
 \hline
 \Gamma \uparrow \vdash
 \end{array}$$

712 But, since LKF^\perp is complete for all singleton subselection, it is complete if one selects only A^\perp
 713 in $E[\underline{A^\perp}]$ and only A in $F[\underline{A}]$. Consequently, we know how to eliminate the cut above. ◀

714 **B** Other complete instances

715 **B.1** Superdeduction

We can go a step further than what is done concerning Deduction Modulo Theory and benefit from focusing to decompose the right-hand side formula after an unfolding has occurred. This leads to what [5] called Superdeduction. [27] studied the links between Superdeduction and focusing, but not with the idea that the rules themselves should be considered as polarized formulas. To link Superdeduction with LKF^\perp , we just need to change the translation of rewriting rules in order to ensure that the right-hand side is decomposed as much as possible. This is done by suppressing the positive delay δ^+ and trying to stay in synchronous (i.e. focused) phase by using negative connectives, until we reach a \exists quantifier, after which we try

to stay in the asynchronous phase. Note, however, that literals are always given a positive polarization. We introduce the negative translation of a formula:

$$\begin{aligned} \llbracket L \rrbracket &= L && \text{when } L \text{ is } \top, \perp \text{ or a literal} \\ \llbracket A \wedge B \rrbracket &= \llbracket A \rrbracket \wedge^- \llbracket B \rrbracket && \llbracket A \vee B \rrbracket = \llbracket A \rrbracket \vee^- \llbracket B \rrbracket \\ \llbracket \exists x. A \rrbracket &= \exists x. \llbracket A \rrbracket && \llbracket \forall x. A \rrbracket = \forall x. \llbracket A \rrbracket \end{aligned}$$

716 and the translation of rewrite rules becomes:

$$717 \quad \llbracket P \rightarrow^- A \rrbracket = \overline{\forall x. \neg P} \vee^- \llbracket A \rrbracket$$

$$718 \quad \llbracket P \rightarrow^+ A \rrbracket = \overline{\forall x. P} \vee^- \llbracket A^\perp \rrbracket$$

720 The synthetic rules given by the translation of rewriting rules correspond exactly to the
721 superrules of Superdeduction.

722 Note that the same kind of encodings can be used to show that Definitional reflection, as
723 defined by [32], can be seen as an instance of LKF^\perp .