EasyChair Preprint

№ 12142

Reducibility Constraints in Superposition

Marton Hajdu, Laura Kovács, Michael Rawson and Andrei Voronkov

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

February 15, 2024

Reducibility Constraints in Superposition

Márton Hajdu[1], Laura Kovács[1], Michael Rawson[1], and Andrei Voronkov[1,2]

[1] TU Wien
[2] U. Manchester and EasyChair

Abstract. Modern superposition inference systems aim at reducing the search space by introducing redundancy criteria on clauses and inferences. This paper focuses on reducing the number of superposition inferences with a single clause by blocking inferences into some terms, provided there were previously made inferences of a certain form performed with predecessors of this clause. Other calculi based on blocking inferences, for example basic superposition, rely on variable abstraction or equality constraints to express irreducibility of terms, resulting however in blocking inferences with all subterms of the respective terms. Here we introduce reducibility constraints in superposition to enable a more expressive blocking mechanism for inferences. We show that our calculus remains (refutationally) complete and present redundancy notions. Our implementation in the theorem prover Vampire demonstrates a considerable reduction in the size of the search space when using our new calculus.

Keywords: Saturation · Superposition · Redundancy· Reducibility constraints

1 Introduction

Automated reasoners in first-order logic with equality commonly rely on the *superposition calculus* [24,5]. This calculus has been extended with various improvements in order to reduce the search space. For instance, avoiding superposition into variables and ordering literals and clauses are common practices in modern theorem provers [20,28,31].

To reduce generation of redundant clauses in equational reasoning, the "basicness" restriction [15] was introduced at the term level. This idea aided, for example, in finding the proof of the Robbins problem [23]. This restriction blocks superposition (rewriting) inferences into terms resulting from (quantifier) instantiations, considering such terms irreducible in further proof steps. This approach was further generalised to block superposition into terms above variable positions in basic superposition/paramodulation [7,25], while preserving refutational completeness. However, blocking and applying different rewrite steps among equal terms impacts proof search. In this paper, we propose a number of different ways to block inferences, so that the resulting calculus remains complete. The effect of these restrictions resembles some strategies from term rewriting, such as innermost and outermost strategies.

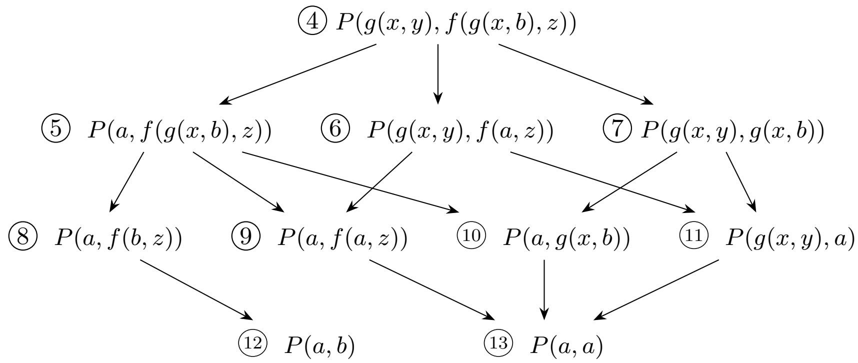

Fig. 1. Possible superposition sequences into ④.

Motivating example. Consider the following satisfiable set $\mathcal{C}$ of clauses:

$$\mathcal{C} = \left\{ \begin{array}{ll} ① \; g(x,b) \simeq a, & ② \; f(x,b) \simeq x, \\ ③ \; g(a,x) \simeq x, & ④ \; P(g(x,y), f(g(x,b),z)) \end{array} \right\}$$

where x, y are variables, a, b constants, f, g function symbols, and P is a predicate symbol. In this paper $\simeq$ denotes equality. Figure 1 shows some derivations of $P(a,a)$ by consecutively superposing into ④ with ① and ②. It also shows a derivation of $P(a,b)$ by superposing into ④ with ①, then with ③ and ②. Note that Figure 1 contains many redundant clauses. For example, ④ is redundant w.r.t. ⑥ and ①, as it is a logical consequence of (smaller) ⑥ and ①. Similarly, ⑦ is redundant w.r.t. ⑪ and ①.

Many derivations of Figure 1 could however be avoided by using a rewrite order between the inferences. For example, a *leftmost-innermost rewrite order* on inferences derives ⑬ along the path ④–⑤–⑨–⑬. Whenever we would deviate from the leftmost-innermost rewrite order when rewriting a term t, we enforce the order by requiring that any term t' that is to the left of or inside t is irreducible in further derivations. In other words, we block further inferences with t'. With such a restriction, we cannot rewrite $g(x,y)$ in clause ⑥, as $g(x,y)$ was to the left of the previously rewritten term $f(g(x,b),z)$. Hence, when using a leftmost-innermost rewrite upon in Figure 1, ⑨ is only generated by the derivation path ④–⑤–⑨. Similarly, ⑪ cannot be derived from ⑦ but can be derived from ⑥.

Our contributions. We introduce a new superposition calculus that enables various ways to block (rewrite) inferences during proof search. Key to our calculus are *reducibility constraints* to restrict the order of superposition inferences (Section 3). Our approach supports and generalizes, among others, the leftmost-innermost rewrite orders mentioned in the motivating example by means of *irreducibility constraints*, allowing us to reduce the number of generated clauses. Furthermore, in our motivating example the derivation ⑤–⑧–⑫ of Figure 1 is not needed for the following reason. By superposing into ② with ③, we derive

$a \simeq b$, which makes one of 13 and 12 redundant w.r.t. the other. As 1 was used to rewrite $g(x, b)$ in Figure 1 and derive 5, we block superposition into $g(x, b)$ with all clauses except 1 in further derivations. We express this requirement via a *one-step reducibility constraint* (Definition 1), resulting in the BLINC – BLocked INference Calculus. As such, BLINC is parameterized by a rewrite ordering and (ir)reducibility constraints.

We prove[3] that our BLINC calculus is refutationally complete, for which we use a model construction technique (Section 4) with new features introduced to take care of constraints. We extend our calculus with redundancy elimination (Section 5). When evaluating the BLINC calculus implemented in the Vampire theorem prover, our experiments show that reducibility constraints significantly reduce the search space (Section 6).

2 Preliminaries

We work in standard *first-order logic with equality*, where equality is denoted by $\simeq$. We use variables x, y, z, v, w and terms s, t, u, l, r, all possibly with indices. A term is *ground* if it contains no variables. A literal is an unordered pair of terms with polarity, i.e. an equality $s \simeq t$ or a disequality $s \not\simeq t$. We write $s \bowtie t$ for either an equality or a disequality. A *clause* is a multiset of literals. We denote clauses by B, C, D and denote by $\square$ the *empty clause* that is logically equivalent to $\perp$.

An *expression* E is a term, literal or clause. We will also consider as expressions constraints and constrained clauses introduced later. An expression is called *ground* if it contains no variables. We write $E[s]$ to state that the expression E contains a distinguished occurrence of the term s at some position. Further, $E[s \mapsto t]$ denotes that this occurrence of s is replaced with t; when s is clear from the context, we simply write $E[t]$. We say that t is a *subterm* of $s[t]$, denoted by $t \trianglelefteq s[t]$; and a *strict subterm* if additionally $t \neq s[t]$, denoted by $t \triangleleft s[t]$. A *substitution* σ is a mapping from variables to terms, such that the set of variables $\{x \mid \sigma(x) \neq x\}$ is finite. We denote substitutions by θ, σ, ρ, μ, η. The application of a substitution θ on an expression E is denoted $E\theta$. A substitution θ is called *grounding for an expression* E if $E\theta$ is ground. We denote the set of grounding substitutions for an expression E by GSubs, that is $\mathsf{GSubs}(E) = \{\theta \mid E\theta \text{ is ground}\}$. We denote the empty substitution by ε.

A substitution θ is more general than a substitution σ if $\theta\eta = \sigma$ for some substitution η. A substitution θ is a *unifier* of two terms s and t if $s\theta = t\theta$, and is a *most general unifier*, denoted $\mathsf{mgu}(s, t)$, if for every unifier η of s and t, there exists a substitution μ s.t. $\eta = \theta\mu$. Recall that the most-general unifiers of terms are idempotent [2].

A binary relation $\rightarrow$ over the set of terms is a *rewrite relation* if (i) $l \rightarrow r \Rightarrow l\theta \rightarrow r\theta$ and (ii) $l \rightarrow r \Rightarrow s[l] \rightarrow s[r]$ for any term l, r, s and substitution θ. The *reflexive and transitive closure* of a relations $\rightarrow$ is denoted by $\rightarrow^*$. We

[3] detailed proofs are in the Appendix

write $\leftarrow$ to denote the inverse of $\rightarrow$. Two terms are *joinable*, denoted by $s \downarrow t$, if $s \rightarrow^* u \leftarrow^* t$. A *rewrite system* R is a set of rewrite rules. A term l is *irreducible* in R if there is no r s.t. $l \rightarrow r \in R$. Joinability w.r.t. R will be denoted by $s \downarrow_R t$. A *rewrite ordering* is a strict rewrite relation. A *reduction ordering* is a well-founded rewrite ordering. In this paper we consider reduction orderings which are total on ground terms, that is they satisfy $s \rhd t \Rightarrow s \succ t$; such orderings are also called *simplification orderings*.

We use the standard definition of a *bag extension* of an ordering [12]. An ordering $\succ$ on terms induces an ordering on literals, by identifying $s \simeq t$ with the multiset $\{s, t\}$ and $s \not\simeq t$ with the multiset $\{s, s, t, t\}$, and using the bag extension of $\succ$. We denote this induced ordering on literals also with $\succ$. Likewise, the ordering $\succ$ on literals induces the ordering on clauses by using the bag extension of $\succ$. Again, we denote this induced ordering on clauses also with $\succ$. The induced relations $\succ$ on literals and clauses are well-founded (resp. total) if so is the original relation $\succ$ on terms. In examples used in this paper, we assume a KBO simplification ordering with constant weight [18].

Many first-order theorem provers work with clauses [28,31,20]. Let S be a set of clauses. Often, systems *saturate* S by computing all logical consequences of S with respect to a sound inference system $\mathcal{I}$. The process of saturating S is called *saturation*. An inference system $\mathcal{I}$ is a set of inference rules of the form

$$\frac{C_1 \ldots C_n}{C},$$

where $C_1, \ldots, C_n$ are the *premises* and C is the *conclusion* of the inference. The inference rule is *sound* if its conclusion is the logical consequence of its premises, that is $C_1, \ldots, C_n \models C$. The inference is *reductive* w.r.t. an ordering $\succ$ if $C \succ C_i$, for some $i = 1, \ldots, n$. An inference system $\mathcal{I}$ is *sound* if all its inferences are sound. An inference system $\mathcal{I}$ is *refutationally complete* if for every unsatisfiable clause set S, there is a derivation of the empty clause in $\mathcal{I}$. An interpretation I is a model of an expression E if E is true in I. A clause C that is false in an interpretation I is a *counterexample* for I. If a clause set contains a counterexample, then it also contain a minimal counterexample w.r.t. a simplification ordering $\succ$ [6].

3 Reducibility Constraints

This section presents a new blocking calculus, called BLINC (BLocked INference Calculus). This calculus uses specific constraints to block inferences.

Definition 1 (Constraints). Let l be a non-variable term and r a term. We call the expression $l \rightsquigarrow r$ a *one-step reducibility constraint*, and the expression $\downarrow l$ an *irreducibility constraint*. A *constraint* is one of the two.

Now let us define the semantics of these constraints.

$$(\mathsf{Sup}_{\Supset})\ \frac{\underline{l \simeq r} \lor C \mid \Pi \quad \underline{s[u] \bowtie t} \lor D \mid \Gamma}{(s[r] \bowtie t \lor C \lor D)\sigma \mid \Delta} \text{where} \quad \begin{array}{l} (1)\ u \text{ is not a variable,} \\ (2)\ \sigma = \mathsf{mgu}(l, u), \\ (3)\ t\sigma \not\succeq s\sigma,\ r\sigma \not\succeq l\sigma, \\ {}^{*}(4)\ \Delta = \Gamma\sigma \cup \mathcal{B}_{\Supset}(s\sigma, l\sigma) \cup \{l\sigma \rightsquigarrow r\sigma\}, \\ {}^{*}(5) \text{ the conclusion is not blocked,} \end{array}$$

$$(\mathsf{EqRes}_{\Supset})\ \frac{\underline{s \not\simeq t} \lor C \mid \Gamma}{C\sigma \mid \Gamma\sigma} \quad \text{where} \quad \begin{array}{l} (1)\ \sigma = \mathsf{mgu}(s, t), \\ {}^{*}(2) \text{ the conclusion is not blocked,} \end{array}$$

$$(\mathsf{EqFac}_{\Supset})\ \frac{\underline{s \simeq t} \lor \underline{u \simeq w} \lor C \mid \Gamma}{(s \simeq t \lor t \not\simeq w \lor C)\sigma \mid \Gamma\sigma} \quad \text{where} \quad \begin{array}{l} (1)\ \sigma = \mathsf{mgu}(s, u), \\ (2)\ t\sigma \not\succeq s\sigma,\ w\sigma \not\succeq t\sigma, \\ {}^{*}(3) \text{ the conclusion is not blocked.} \end{array}$$

Fig. 2. The BLINC calculus

Definition 2 (Satisfied Constraints, Violated Constraints). Let R be a rewrite system. We say that R *satisfies* $l \rightsquigarrow r$ if $l \to r \in R$ and *satisfies* $\downarrow\!l$ if l is irreducible in R. We say that R *violates* a constraint if it does not satisfy it.

An expression $C \mid \Gamma$ is a *constrained clause*, where C is a clause and Γ a finite set of constraints. We denote constrained clauses $\mathcal{C}$, $\mathcal{D}$, possibly with indices.

Definition 3 (Blocked Constrained Clause, Blocked Inference). Let $\mathcal{C} = C \mid \Gamma$ be a constrained clause. We call the constraint $l \rightsquigarrow r \in \Gamma$ *active in* $\mathcal{C}$ if $s \succ l$ for some term s in C. Likewise, we call $\downarrow\!l \in \Gamma$ *active in* $\mathcal{C}$ if $s \succ l$ for some term s in C. We call $\mathcal{C}$ *blocked* if either it contains two active constraints $l \rightsquigarrow r$ and $l \rightsquigarrow r'$ such that r and r' are not unifiable, or it contains two active constraints $l \rightsquigarrow r$ and $\downarrow\!l$. An inference is *blocked* if its conclusion is blocked.

Our superposition calculus BLINC uses constrained clauses and bans inferences with blocked conclusions. For that, we attach constraints to clauses, as follows.

Definition 4 (S-ordering). An *S-ordering* is a partial strict well-order $\Supset$ on terms, stable under substitutions. For that, we use the function $\mathcal{B}_{\Supset}$ defined below to attach constraints to clauses.

$$\mathcal{B}_{\Supset}(s, l) := \{\downarrow\!u \mid u \Subset l, u \text{ is non-variable and } u \trianglelefteq s\}$$

BLINC is shown in Figure 3. We assume a literal selection function satisfying the standard condition on $\succ$ and underline selected literals. The next example illustrates blocked BLINC inferences.

Example 1. We use the order $\succ$ on terms as the S-ordering. A $\mathsf{Sup}_{\Supset}$ inference of BLINC into 4 with 2 from our motivating example from page 2 results in

$$\frac{f(x, b) \simeq x \quad P(g(x, y), f(g(x, b), z))}{P(g(x, y), g(x, b)) \mid \{\downarrow\!b, \downarrow\!g(x, y), \downarrow\!g(x, b), f(g(x, b), b) \rightsquigarrow g(x, b)\}}\ .$$

Note that the conclusion constrains clause 7 of Figure 1. Now, the superposition of 1 into $g(x, y)$, and hence clause 11 of Figure 1, is blocked:

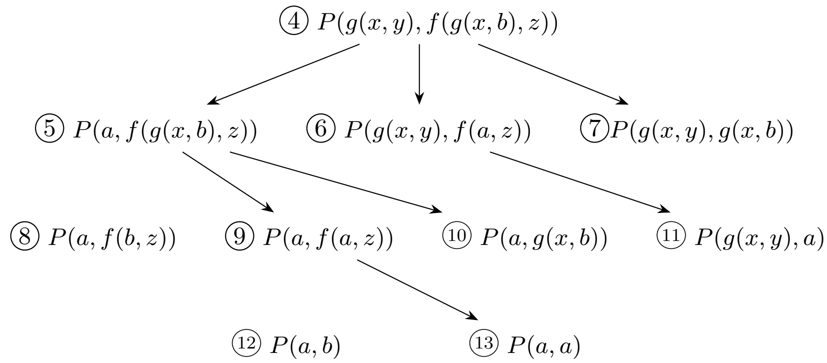

Fig. 3. Inferences from Figure 1 with blocked inferences in BLINC removed. Figure 3 illustrates the effectiveness of blocking constraints when compared to Figure 1: we removed arcs corresponding to inferences blocked when the order $\succ$ is used as the S-ordering. Of the 14 original inferences as in Figure 1, only 7 are non-blocked in Figure 3.

$$\frac{g(x,b) \simeq a \quad P(g(x,y), g(x,b)) \mid \{\downarrow g(x,y), \downarrow g(x,b), f(g(x,b),b) \rightsquigarrow g(x,b)\}}{P(a, g(x,b)) \mid \{\downarrow b, \downarrow g(x,b), f(g(x,b),b) \rightsquigarrow g(x,b), g(x,b) \rightsquigarrow a\}}$$

Note that the conclusion blocks clause ⑪ of Figure 1 by the active constraints $\downarrow g(x,b)$ and $g(x,b) \rightsquigarrow a$. Figure 3 shows the modified version of Figure 1, when using the blocking inferences of BLINC to generate less clauses than in Figure 1. □

Example 2. Consider now a sequence of superposition inferences into ④ by ① and then by ③. That is, we consider the derivation ④–⑤–⑧ from Figure 1 as:

$$\frac{g(a,x) \simeq x \quad \dfrac{g(x,b) \simeq a \quad P(g(x,y), f(g(x,b),z))}{P(a, f(g(x,b),z)) \mid \{\downarrow b, g(x,b) \rightsquigarrow a\}}}{P(a, f(b,z)) \mid \{\downarrow a, \downarrow b, g(a,b) \rightsquigarrow a, g(a,b) \rightsquigarrow b\}}$$

The resulting conclusion is constrained and blocked, as we have two active constraints $g(a,b) \rightsquigarrow a$ and $g(a,b) \rightsquigarrow b$. As such and as shown in Figure 3, clause ⑨ (and also clause ⑫) will not be generated by BLINC, in contrast to Figure 1. □

4 Model Construction in BLINC

This section shows completeness of BLINC, with a proof which resembles that of Duarte and Korovin [13]. We start by adjusting terminology to our setting and discussing key differences compared to standard completeness proofs.

Definition 5 (Closure). Let $\mathcal{C} = C \mid \Gamma$ be a constrained clause and θ a substitution. The pair $\mathcal{C} \cdot \theta$ is called a *closure* and is logically equivalent to $C\theta$. A closure $(C \mid \Gamma) \cdot \theta$ is *ground* if $C\theta \mid \Gamma\theta$ is ground, in which case we say that θ is *grounding* for $C \mid \Gamma$ and call $(C \mid \Gamma) \cdot \theta$ a *ground instance of* $C \mid \Gamma$. Note that a ground instance of a constrained clause is a closure.

The set of all ground instances of $\mathcal{C}$ is denoted $\mathcal{C}^*$. We will denote ground closures by $\mathbb{C}, \mathbb{D}$, maybe with indexes. If N is a set of constrained clauses, then N^* is defined as $\bigcup_{\mathcal{C} \in N} \mathcal{C}^*$. If $C \succ D$, we write $C \mid \Gamma \succ D \mid \Delta$. Similarly, if $C\theta \mid \Gamma\theta \succ D\sigma \mid \Delta\sigma$, then we write $(C \mid \Gamma) \cdot \theta \succ (D \mid \Delta) \cdot \sigma$.

A crucial part in the completeness proof of BLINC is reducing minimal counterexamples to smaller ones. However, due to blocked inference conditions (5) in $\mathsf{Sup}_{\Supset}$, (2) in $\mathsf{EqRes}_{\Supset}$, and (3) in $\mathsf{EqFac}_{\Supset}$, such a counterexample reduction may not be possible. We solve this problem in three steps:

1. Given a saturated set of clauses N, we construct a model for a subset of its closures $\mathcal{U}(N) \subseteq N^*$, namely, for so-called *unblocked closures* that do not block any reductive inferences with smaller clauses (Definition 6).
2. We show that if the empty clause $\square$ is not in $\mathcal{U}(N)$, then the model satisfies each closure in $\mathcal{U}(N)$ (Theorem 1). That is, we show that counterexamples in $\mathcal{U}(N)$ can be reduced to smaller counterexamples, which are also in $\mathcal{U}(N)$. This avoids the aforementioned problem with blocked inferences.
3. We then show that the model also satisfies all closures in N^* (Theorem 2).

Definition 6 (Partial/Total Models, Blocked/Productive Closures). Let N be a set of constrained clauses. We will define, for every closure $\mathbb{C} \in N^*$, the rewrite system $R_{\mathbb{C}}$ and refer to all such rewrite systems as *partial models*. The definition will be made by induction on the relation $\succ$ on ground closures. In parallel to defining $R_{\mathbb{C}}$, we also define the rewrite system

$$R_{\prec\mathbb{C}} = \bigcup_{\mathbb{D} \prec C} R_{\mathbb{D}}.$$

The partial model $R_{\mathbb{C}}$ will either be the same as $R_{\prec\mathbb{C}}$, or obtained from $R_{\prec\mathbb{C}}$ by adding a single rewrite rule. In the latter case will call the closure $\mathbb{C}$ *productive*.

The *reduced closure* of a ground closure $\mathcal{C} \cdot \theta$ is defined as the closure $\mathcal{C} \cdot \theta'$ such that for each variable x occurring in $\mathcal{C}$, we have that $\theta'(x)$ is the normal form of $\theta(x)$ in $R_{\prec\mathcal{C}\cdot\theta}$. We call a ground closure *reduced* if its reduced form coincides with this closure. Let $\mathcal{C} \cdot \theta$ be a ground closure and $\mathcal{C} \cdot \theta'$ be its reduced form. We say that $\mathcal{C} \cdot \theta$ is *blocked w.r.t.* N if $R_{\prec\mathcal{C}\cdot\theta'}$ violates some constraint in $\Gamma\theta'$ that is active in $\mathcal{C}\theta'$. A closure that is not blocked w.r.t. N is called *unblocked w.r.t.* N. Let $\mathcal{C} = (l \simeq r \vee C') \mid \Gamma$. The closure $\mathcal{C} \cdot \theta$ is called *productive* if

(i) $\mathcal{C} \cdot \theta$ is false in $R_{\prec\mathcal{C}\cdot\theta}$,
(ii) $l\theta \simeq r\theta$ is strictly maximal in $\mathcal{C}\theta$,
(iii) $l\theta \succ r\theta$,
(iv) $C'\theta$ is false in $R_{\prec C\cdot\theta} \cup \{l\theta \to r\theta\}$,
(v) $l\theta$ is irreducible in $R_{\prec\mathcal{C}\cdot\theta}$,
(vi) $\mathcal{C} \cdot \theta$ is unblocked w.r.t N.

Now we define

$$R_{\mathcal{C}\cdot\theta} = \begin{cases} R_{\prec\mathcal{C}\cdot\theta} \cup \{l\theta \to r\theta,\} & \text{if } \mathcal{C}\cdot\theta \text{ is productive;} \\ R_{\prec\mathcal{C}\cdot\theta}, & \text{otherwise.} \end{cases}$$

$$R_\infty = \bigcup_{\mathbb{C}\in N^*} R_{\mathbb{C}}$$

Finally, we call R_∞ *the total model* and define $\mathcal{U}(N)$ as the set of all closures in N^* unblocked w.r.t. N. □

This construction has two standard properties that we will use in our proofs:

1. $R_{\mathbb{C}} \models \mathbb{C}$ if and only if for all $\mathbb{D} \succ \mathbb{C}$ we have $R_{\mathbb{D}} \models \mathbb{C}$, if and only if $R_\infty \models \mathbb{C}$.
2. R_∞ is non-overlapping, terminating and hence canonical.

The crucial difference between our model construction and the standard model construction is the condition on productive closures to be unblocked w.r.t. N. Let us now define our redundancy notions based on $\mathcal{U}(N)$ as follows.

Definition 7 (Redundant Clause/Inference). A constrained clause $\mathcal{C}$ is *redundant w.r.t.* N if every ground instance of $\mathcal{C}$ is either blocked w.r.t. N, or follows from smaller ground closures in $\mathcal{U}(N)$. An inference $\mathcal{C}_1, ..., \mathcal{C}_n \vdash \mathcal{D}$ is *redundant w.r.t.* N if for each θ grounding for $\mathcal{C}_1, \dots, \mathcal{C}_n$ and $\mathcal{D}$ either

(i) one of $\mathcal{C}_1\cdot\theta, ..., \mathcal{C}_n\cdot\theta, \mathcal{D}\cdot\theta$ is blocked w.r.t. N, or
(ii) $\mathcal{D}\cdot\theta$ follows from the set of ground closures $\{\mathbb{C} \mid \mathbb{C} \in \mathcal{U}(N) \text{ and } \mathcal{C}_i\cdot\theta \succ \mathbb{C} \text{ for some } i\}$. □

Definition 8 (Saturation up to Redundancy). A set of constrained clauses N is *saturated up to redundancy* if, given non-redundant constrained clauses $\mathcal{C}_1, ..., \mathcal{C}_n \in N$, any BLINC inference $\mathcal{C}_1, ..., \mathcal{C}_n \vdash \mathcal{D}$ is redundant w.r.t. N.

From now on, let N be an arbitrary but fixed set of constrained clauses. We will formulate a sequence of lemmas used in the completeness proof, whose proofs are included in the appendix. The following lemma is used to show that unary inferences with an unblocked premise result in an unblocked conclusion.

Lemma 1. (Unblocking Inferences) *Suppose $\mathcal{C}, \mathcal{D} \in N$ and θ and σ are substitutions irreducible in $R_{\prec\mathcal{C}\cdot\theta}$ and in $R_{\prec\mathcal{D}\cdot\sigma}$, respectively. If $\mathcal{C}\cdot\theta \succ \mathcal{D}\cdot\sigma$, $\Gamma\theta \supseteq \Delta\sigma$ and $\mathcal{C}\cdot\theta$ is unblocked w.r.t. N, then $\mathcal{D}\cdot\sigma$ is unblocked w.r.t. N.*

We next prove that the conclusion of a blocked inference is redundant, that is, the conditions for blocking inferences in BLINC are correct.

Lemma 2. (Redundancy with Blocked Clauses) *Let $\mathcal{C}$ be a constrained clause. If $\mathcal{C}$ is blocked, then all ground instances of $\mathcal{C}$ are blocked w.r.t. N.*

The next lemma resembles the standard lemma on counterexample reduction.

Lemma 3 (Unblocked $\mathsf{Sup}_\Supset$). *Suppose that (a) $\mathcal{D} = \underline{s \bowtie t} \vee D \mid \Gamma$ is a constrained clause in N, (b) $\mathcal{D}\cdot\theta$ a ground closure unblocked w.r.t. N, (c) θ is irreducible in $R_{\prec\mathcal{D}\cdot\theta}$, (d) $s\theta \succeq t\theta$, (e) $s\theta$ is reducible in $R_{\prec\mathcal{D}\cdot\theta}$.*

Then there exist a constrained clause $(\underline{l \simeq r} \vee C \mid \Pi) \in N$, a $\mathsf{Sup}_\Supset$-inference

$$(\mathsf{Sup}_{\ni})\ \frac{\underline{l \simeq r} \vee C \mid \Pi \quad \underline{s[u]} \bowtie t \vee D \mid \Gamma}{(s[r] \bowtie t \vee C \vee D)\sigma \mid \Delta}$$

and a substitution τ such that (i) $\mathcal{D}\sigma\tau = \mathcal{D}\theta$, (ii) $\underline{l \simeq r} \vee C \mid \Pi \cdot \sigma\tau$ is productive, and $(s[r] \bowtie t \vee C \vee D)\sigma \mid \Delta \cdot \sigma\tau$ is unblocked w.r.t. N.

We are now ready to show completeness of BLINC, starting with the following.

Theorem 1 (Model of $\mathcal{U}(N)$). *Let N be saturated up to redundancy and $\square \notin N$. Then for each $\mathbb{C} \in \mathcal{U}(N)$ we have $R_{\mathbb{C}} \models \mathbb{C}$.*

When $R_{\mathbb{C}} \models \mathbb{C}$, we will simply say that $\mathbb{C}$ is true. Note that this implies that $R_{\mathbb{D}} \models \mathbb{C}$ for all $\mathbb{D} \succeq \mathcal{C}$, and also $R_\infty \models \mathbb{C}$. We say that $\mathbb{C}$ is false if it not true.

Proof. Here, we only prove a few representative cases and refer to the Appendix for complete argumentation. Assume, by contradiction, that $\mathcal{U}(N)$ contains a ground closure $\mathbb{C}$ such that $R_{\mathbb{C}} \not\models \mathbb{C}$. Since $\succ$ is well-founded, then N^* contains a minimal non-blocked closure $\mathcal{C} \cdot \theta$ such that $R_{\mathcal{C}\cdot\theta} \not\models \mathcal{C} \cdot \theta$.

Case 1. *$\mathcal{C}$ is redundant w.r.t. N.* The closure $\mathcal{C} \cdot \theta$ is non-blocked, so it follows from smaller closures $\mathcal{C}_1, \ldots, \mathcal{C}_n$ in $\mathcal{U}(N)$. Then there is some $\mathcal{C}_i$ which is false too, and we are done. $\square$

Case 2. *$\mathcal{C}$ contains a variable x such that $x\theta$ is reducible in $R_{\prec\mathcal{C}\cdot\theta}$.* The reduced closure $\mathcal{C}\cdot\theta'$ of $\mathcal{C}\cdot\theta$ is unblocked w.r.t. N, so $\mathcal{C}\cdot\theta' \in \mathcal{U}(N)$. Since $x\theta \succ x\theta'$ and for all variables y different from x we have $y\theta \succeq y\theta'$, we have $\mathcal{C}\cdot\theta \succ \mathcal{C}\cdot\theta'$, then $\mathcal{C}\cdot\theta'$ is true. Since $y\theta = y\theta'$ is true in R_∞ for all variables y, we also have that $\mathcal{C}\cdot\theta'$ is equivalent to $\mathcal{C}\cdot\theta$ in R_∞, hence $\mathcal{C}\cdot\theta$ is true and we obtain a contradiction. $\square$

Case 3. *There is a reductive inference $\mathcal{C}_1, \ldots, \mathcal{C}_n \vdash \mathcal{D}$ with $\mathcal{C}_1, \ldots, \mathcal{C}_n \in N$ which is redundant w.r.t. N such that (a) $\{\mathcal{C}_1 \cdot \theta, \ldots, \mathcal{C}_n \cdot \theta\} \subseteq \mathcal{U}(N)$, (b) $\mathcal{D} \cdot \theta$ is unblocked w.r.t. N, (c) $\mathcal{C} \cdot \theta = \max\{\mathcal{C}_1 \cdot \theta, \ldots, \mathcal{C}_n \cdot \theta\}$, and (d) $\mathcal{D} \cdot \theta \models \mathcal{C} \cdot \theta$.* $\mathcal{D} \cdot \theta$ is implied by ground closures in $\mathcal{U}(N)$ smaller than $\mathcal{C} \cdot \theta$. These ground closures are then true in $R_{\mathcal{C}\cdot\theta}$, so $\mathcal{D}\cdot\theta$ is true, and hence $\mathcal{C}\cdot\theta$ is also true in $R_{\mathcal{C}\cdot\theta}$, contradiction. $\square$

Case 4. *None of the previous cases apply, and a negative literal $s \not\simeq t$ is selected in $\mathcal{C}$, i.e. $\mathcal{C} = \underline{s \not\simeq t} \vee C \mid \Gamma$.* $\mathcal{C} \cdot \theta$ is false in $R_{\mathcal{C}\cdot\theta}$, so $s\theta \downarrow_{R_{\mathcal{C}\cdot\theta}} t\theta$. W.l.o.g., assume $s\theta \succeq t\theta$.

Subcase 4.1. *$s\theta = t\theta$, where s and t are unifiable.* Consider the $\mathsf{EqRes}_{\ni}$ inference

$$\frac{\underline{s \not\simeq t} \vee C \mid \Gamma}{C\sigma \mid \Gamma\sigma}$$

where $\sigma = \mathsf{mgu}(s,t)$. Take any ground instance $\mathcal{D} \cdot \rho = (C\sigma \mid \Gamma\sigma) \cdot \rho$ such that $\sigma\rho = \theta$; by the idempotence of σ, we have $\mathcal{D} \cdot \rho = \mathcal{D} \cdot \theta$. Clearly, $\mathcal{C} \cdot \theta \succ \mathcal{D} \cdot \theta$ and $\mathcal{D} \cdot \theta$ implies $\mathcal{C} \cdot \theta$. As $\mathcal{C} \cdot \theta \succ \mathcal{D} \cdot \theta$ and $\Gamma\sigma\rho = \Gamma\sigma\theta = \Gamma\theta$, Lemma 1 implies that $\mathcal{D} \cdot \theta$ is unblocked w.r.t. N. By Case 1, $\mathcal{D}$ is not redundant, hence $\mathcal{D} \in N$. But then $\mathcal{D} \cdot \theta$ is a false closure in $\mathcal{U}(N)$, which is strictly smaller than $\mathcal{C} \cdot \theta$, so we obtain a contradiction. $\square$

Subcase 4.2. $s\theta \succ t\theta$. By conditions on the literal selection, we assume that $s\theta \succ t\theta$ is maximal in $\mathcal{C}$. By Lemma 3, there is a $\mathsf{Sup}_{\ni}$ inference into $s\theta$ with a ground closure such that the result $\mathcal{C}' \cdot \theta$ is unblocked w.r.t. N. This closure is of the form $\mathcal{D} \cdot \theta = (l \simeq r \vee D \mid \Pi) \cdot \theta$ and we have the following $\mathsf{Sup}_{\Supset}$ inference

$$\frac{\underline{l \simeq r} \vee D \mid \Pi \quad \underline{s[l'] \not\simeq t} \vee C \mid \Gamma}{(s[r] \not\simeq t \vee C \vee D)\sigma \mid \Delta}$$

where $\sigma = \mathsf{mgu}(l, l')$. Note that $\mathcal{C}' = s[r] \not\simeq t \vee C \vee D$ and $\mathcal{C}' \cdot \rho = \mathcal{C}' \cdot \theta$. Then, $\mathcal{C} \cdot \theta \succ \mathcal{C}' \cdot \theta$ and $\mathcal{D} \cdot \theta$ and $\mathcal{C}' \cdot \theta$ imply $\mathcal{C} \cdot \theta$. Since $\mathcal{C}' \cdot \theta$ is unblocked w.r.t. N, using Lemma 2, we get that $\mathcal{C}'$ is not blocked w.r.t. N, and condition (5) of $\mathsf{Sup}_{\Supset}$ is satisfied. Similar to Case 4.1, we have that the conclusion is a smaller false unblocked closure, so we obtain a contradiction. □

Next we show that for a saturated set of clauses N, if R_∞ is a model for $\mathcal{U}(N)$, then it is also a model of N^*, that is, R_∞ satisfies also all blocked closures in N^*. This follows from the next theorem.

Theorem 2 (Model of N^*). *Let N be a saturated set of clauses. Every blocked closure $\mathcal{C} \cdot \theta \in N^*$ follows from $\mathcal{U}(N)$.*

Using Theorems 1–2, we obtain completeness of BLINC.

Corollary 1 (Completeness of BLINC). *Let N be saturated up to redundancy. If N does not contain □, then N is satisfiable.*

We conclude with a remark on *constraint inheritance* in BLINC. Note that in the $\mathsf{Sup}_{\Supset}$ inference rule of Figure 3, constraints are inherited only from the right premise. It is possible to block more inferences without losing refutational completeness of BLINC, by allowing constraint inheritance from the left premise in the $\mathsf{Sup}_{\Supset}$ rule as well. However, we cannot propagate constraints that are non-active in the left premise, as they may become active in the conclusion, making the inference blocked. This effect is illustrated in the following example.

Example 3. Consider a superposition into 1 with 3

$$\frac{g(x,b) \simeq a \quad g(a,x) \simeq x}{a \simeq b \mid \{\downarrow a, \downarrow b, g(a,b) \rightsquigarrow a\}}$$

If $b \succ a$, then $\downarrow a$ is the only active constraint in the conclusion. Consider a superposition with 4 where constraints are inherited from both premises:

$$\frac{a \simeq b \mid \{\downarrow a, \downarrow b, g(a,b) \rightsquigarrow a\} \quad P(g(x,y), f(g(x,b),z))}{P(g(x,y), f(g(x,a),z)) \mid \{\downarrow a, \downarrow b, g(a,b) \simeq a, b \rightsquigarrow a\}}$$

In the conclusion, $\downarrow b$ and $b \rightsquigarrow a$ are both active, which blocks the inference. □

5 Redundancy Detection in BLINC

In this section we discuss redundancy detection in BLINC. We give sufficient conditions for a clause to be redundant when inferences of a specific form are applied. As usual, we call a *simplifying inference*, or *simplification*, any inference such that one of the premises becomes redundant after the conclusion is added to the current set of clauses. Inference rules whose instances are simplifications are called *simplification rules*. When we display a simplification rule, we will denote clauses that become redundant by drawing a line through them.

Definition 7 gives rise to two kinds of simplification rules: (i) based on blocking and (ii) when one of the premises $\mathcal{C} \cdot \theta$ follows from smaller constrained clauses. The following definition captures the first kind of redundancy.

Definition 9 (Closure/Clause Blocked Relative To Closure/Clause). A ground closure $\mathbb{C}$ is *blocked relative to* a ground closure $\mathbb{D}$ if for every set of constrained clauses N, if $\mathbb{D}$ is blocked w.r.t. N^*, then $\mathbb{C}$ is blocked w.r.t. N^* too. A constrained clause $\mathcal{C}$ is *blocked relative to* a constrained clause $\mathcal{D}$, if every ground instance of $\mathcal{C}$ is blocked relative to some ground instance of $\mathcal{D}$.

This notion will be used for defining simplification rules. We will next present sufficient conditions for checking that a constrained clause is blocked relative to another constrained clause. For example, each ground closure of a clause $C \mid \emptyset$ is unblocked w.r.t. any set N, hence everything is blocked relative to that ground closure. Further, each ground closure with a reducible substitution is blocked relative to its reduced closure.

Definition 10 (Well-Behaved Constrained Clause). Let $\mathcal{C} = C \mid \Gamma$ be a constrained clause. We say that $\mathcal{C}$ is *well-behaved* if (i) all constraints in Γ are active in C, and for each $\gamma \in \Gamma$, (ii) if $\gamma = {\downarrow}l$, then ${\downarrow}u \in \Gamma$ for all $u \lhd l$, and (iii) if $\gamma = l \rightsquigarrow r$, then ${\downarrow}u \in \Gamma$ for all $u \lhd l$ and l contains all variables of r.

Example 4. The clause $P(a, f(b, z)) \mid \{{\downarrow}a, g(a, b) \rightsquigarrow a\}$ is not well-behaved but $P(a, f(b, z)) \mid \{{\downarrow}a, {\downarrow}b, g(a, b) \rightsquigarrow a\}$ is. The clause $a \simeq b \mid \{{\downarrow}a, {\downarrow}b, g(a, b) \rightsquigarrow a\}$ is not well-behaved since it contains constraints not active in the clause. □

Lemma 4. (Relatively Blocked Well-Behavedness) *Let $\mathcal{C} = C \mid \Gamma$ and $\mathcal{D} = D \mid \Delta$ be well-behaved constrained clauses, and σ a substitution. Then $\mathcal{C}$ is blocked relative to $\mathcal{D}$ if $\mathcal{C} \succ \mathcal{D}\sigma$ and $\Gamma \supseteq \Delta\sigma$.*

In the sequel, we assume that each constrained clause is well-behaved. We next adjust two standard simplifications within superposition [14], namely demodulation in Theorem 3 and subsumption in Theorem 4. Our analogue of *demodulation* is the following special case of $\mathsf{Sup}_{\Supset}$ in BLINC:

$$(\mathsf{Dem}_{\Supset})\ \frac{l \simeq r \mid \Delta \quad \cancel{C[l\sigma] \mid \Gamma}}{C[r\sigma] \mid \Gamma} \qquad \text{where} \quad \begin{array}{l}(1)\ l\sigma \succ r\sigma,\\ (2)\ C[l\sigma] \succ (l \simeq r)\sigma,\\ (3)\ \Delta\sigma \subseteq \Gamma.\end{array}$$

Theorem 3. (BLINC Demodulation) $\mathsf{Dem}_{\Supset}$ *is a simplification rule. That is, $C[l\sigma] \mid \Gamma$ is redundant w.r.t. any constrained clause set that contains $l \simeq r \mid \Delta$ and $C[r\sigma] \mid \Gamma$.*

In addition to simplification rules, we will also consider *deletion rules*. These rules delete a (redundant) constrained clause from N provided that N contains another constrained clause or set of constrained clauses. The below deletion rule is our analogue of *subsumption*:

$$(\mathsf{Subs}_{\ni})\ \frac{D \mid \Delta \quad \cancel{C \mid \Gamma}}{} \quad \text{where} \quad \begin{array}{l}(1)\ D\sigma \subsetneq C,\\ (2)\ \Delta\sigma \subseteq \Gamma,\end{array} \text{ for some substitution } \sigma.$$

Theorem 4. **(BLINC Subsumption)** $\mathsf{Subs}_{\ni}$ *is a deletion rule. That is,* $C \mid \Gamma$ *is redundant w.r.t. any constrained clause set that contains* $D \mid \Delta$*.*

We also introduce two deletion rules based on properties of the constraints of a clause. Namely, in Theorem 5 we introduce a deletion rule resembling "basic blocking" [24], whereas Theorem 6 exploits deletion based on rewrite orders. Consider therefore the following rule:

$$(\mathsf{Block}_{\ni})\ \frac{l \simeq r \mid \Delta \quad \cancel{C \mid \Gamma}}{} \quad \text{where} \quad \begin{array}{l}(1)\ C \succ (l \simeq r)\sigma \text{ and } l\sigma \succ r\sigma,\\ (2)\ \Delta\sigma \subseteq \Gamma,\\ (3)\ \text{either (i) } {\downarrow} l\sigma \in \Gamma\\ \quad \text{or (ii) } l\sigma \rightsquigarrow r' \in \Gamma \text{ and } r' \succ r\sigma.\end{array}$$

Theorem 5. **(BLINC Blocking)** $\mathsf{Block}_{\ni}$ *is a deletion rule. That is,* $C \mid \Gamma$ *is redundant w.r.t. any constrained clause that contains* $l \simeq r \mid \Delta$*.*

Our last deletion inference relies on the fact that all rewrite rules in any partial model have to be oriented left-to-right according to $\succ$. That is,

$$(\mathsf{Orient}_{\ni})\ \frac{\cancel{C \mid \Gamma \cup \{l \rightsquigarrow r\}}}{} \quad \text{where} \quad \begin{array}{l}(1)\ r \succ l,\\ (2)\ C \succ (l \simeq r).\end{array}$$

Theorem 6. **(BLINC Orientation)** $\mathsf{Orient}_{\ni}$ *is a deletion rule. That is,* $C \mid \Gamma \cup \{l \rightsquigarrow r\}$ *is redundant w.r.t. any constrained clause set.*

We illustrate the above simplification and deletion rules with the following example.

Example 5. Consider the following well-behaved constrained clauses:

$$\begin{array}{ll}(1)\ P(g(a,x),b) \mid \{{\downarrow}b, f(x,b) \rightsquigarrow b\}, & (2)\ P(g(y,z),w) \mid \{f(z,w) \rightsquigarrow b\}\\ (3)\ g(a,z) \simeq b \mid \{{\downarrow}b\}, & (4)\ f(x,y) \simeq a \mid \emptyset\end{array}$$

By Theorem 4, clause (2) subsumes clause (1). By Theorem 3, clause (1) can be simplified with clause (3) into $P(b,b) \mid \{{\downarrow}b, f(x,b) \rightsquigarrow a\}$. Finally, assuming $b \succ a$, clauses (1) and (2) are redundant w.r.t. clause (4) by Theorem 5. □

Remark 1. **(Simplification Heuristics via Unblocking)** We note that further simplifications (and heuristics) can be implemented by removing constraints from constrained clauses. This process of removing constraints is captured via the following rule:

Variant	UEQ		PEQ	
	Solved	Uniques	Solved	Uniques
`baseline`	778	15	1276	34
`blinc1`	316	0	411	0
`blinc2`	327	0	425	0
`blinc3`	610	0	809	0
`blinc4`	775	13	1270	28

Fig. 4. Experimental comparison using variants BLINC in Vampire, using 1455 UEQ problems and 2422 PEQ problems.

$$(\mathsf{Unblock})\ \frac{\cancel{C \mid \Gamma}}{C \mid \Delta} \quad \text{where} \quad \Delta \subset \Gamma.$$

Clearly, Unblock is a simplification rule, as removing constraints from a constrained clause preserves completeness in BLINC.

We conclude this section by noting that Theorems 3–6 can be adjusted and combined using the ground redundancy of Definition 7. This results in stronger redundancy detection, as the following example illustrates.

Example 6. Consider the following $\mathsf{Sup}_{\ni}$ inference:

$$\frac{g(f(v,w),a) \simeq g(w,a) \mid \emptyset \quad f(g(f(x,y),z),f(y,x)) \simeq z \mid \emptyset}{f(g(y,a),f(y,x)) \simeq a \mid \Delta} \ \sigma = \left\{ \begin{array}{c} v \mapsto x, \\ w \mapsto y, \\ z \mapsto a \end{array} \right\},$$

where $\Delta = \{\downarrow f(x,y), \downarrow f(y,x), \downarrow a, g(f(x,y),a) \rightsquigarrow g(y,a)\}$. Note that the conclusion is a well-behaved constrained clause. The conclusion cannot be simplified by clauses

$$(1)\ f(x,y) \simeq f(y,x) \quad \text{and} \quad (2)\ f(x,x) \simeq x,$$

using any of Theorems 3–6. However, using similar conditions as in the $\mathsf{Block}_{\ni}$ deletion rule, we can do the following. Let θ be a substitution that makes the conclusion ground. By a comparative case distinction on $x\theta$ and $y\theta$,

(i) if $x\theta \succ y\theta$, then using clause (1), by $\downarrow f(x,y) \in \Delta$ and $f(x,y)\theta \succ f(y,x)\theta$;
(ii) if $x\theta = y\theta$, then using clause (2) by $\downarrow f(x,y) \in \Delta$ (or $\downarrow f(y,x) \in \Delta$), $f(x,y)\theta = f(x,x)\theta \succ x\theta$ (or $f(y,x)\theta = f(x,x)\theta \succ x\theta$); and
(iii) if $x\theta \prec y\theta$, then using clause (1) again, by $\downarrow f(y,x) \in \Delta$ and $f(y,x)\theta \succ f(x,y)\theta$;

we conclude that the ground closure $(f(g(y,a),f(y,x)) \simeq a \mid \Delta) \cdot \theta$ is redundant in all cases, hence the conclusion is redundant w.r.t. clauses (1) and (2). □

6 Evaluation

We implemented[4] BLINC in Vampire [20], together with the simplification rules of Section 5. We have also implemented a redundancy check called *orderedness*

[4] `https://github.com/vprover/vampire/commit/9c42b448996947e8`

that eagerly checks if the result of a superposition can be deleted. We experimented with several variants of BLINC with redundancy elimination, using different heuristics for removing constraints from clauses via Unblock: (i) `blinc1` does not use Unblock; (ii) `blinc2` uses Unblock to remove constraints inherited from premises, hence only conclusions of $\mathsf{Sup}_{\ni}$ will contain constraints; (iii) `blinc3` uses Unblock occasionally on the clause that would simplify the most clauses in the search space when unconstrained; (iv) `blinc4` uses Unblock on all clauses at activation. We compare these to standard superposition (`baseline`).

All our experiments use a DISCOUNT saturation loop [11] and a Knuth-Bendix ordering, with a timeout of 100 seconds and without AVATAR [30]. We used benchmarks from TPTP version 8.1.2 [29], in particular all benchmarks from the unit equality (UEQ) and pure equality (PEQ) divisions.

Our experimental results are summarized in Figure 4. The results show that `blinc1` performs poorly compared to `baseline`, `blinc3` and `blinc4`, and that `blinc2` performs only slightly better than `blinc1`. The variant `blinc3` performs much better than `blinc1` and `blinc2` but it is still does not solve any new problems. The variant `blinc4` performs however similarly well to the state-of-the-art `baseline`, while solving also different, 13 unique, benchmarks. Our preliminary results are therefore encouraging for complementing state-of-the-art superposition proving with BLINC reasoning, possibly in a portfolio solver.

We also analysed the impact of BLINC variants on skipping superposition inferences during proof search. Figure 6 shows the distribution of benchmarks by percentage of skipped superposition inferences among all superposition inferences during our runs for `blinc` variants. `blinc1` skips more than half of superposition inferences in a significant number of benchmarks, while the least restrictive `blinc4` still reduces the number of superposition inferences by a significant amount in most benchmarks.

7 Related work

The basicness restriction [26,15] was extended to first-order logic, for example, in *basic superposition* [25] and *basic paramodulation* [7]. The former uses ground unification, the latter closures and variable abstraction to capture irreducibility constraints. In basic paramodulation, *redex orderings* are used similarly to S-orderings in our framework. BLINC expresses more fine-grained blocking, for example, distinguishing between different superpositions on the same term.

Several critical pair criteria in completion-based theorem proving use irreducibility notions. *Blocking* [4] is similar to basicness, while *compositeness* [4,16] forbids any superpositions into terms with reducible subterms. *General superposition* [32,33] avoids superpositions when more general ones or ones symmetric in variables have been performed. Our BLINC framework handles all such restrictions. These criteria are instances of the *connectedness* criterion [3], which has been also explored in *ground joinability* [1], *ground reducibility* [21] and *ground connectedness* [13].

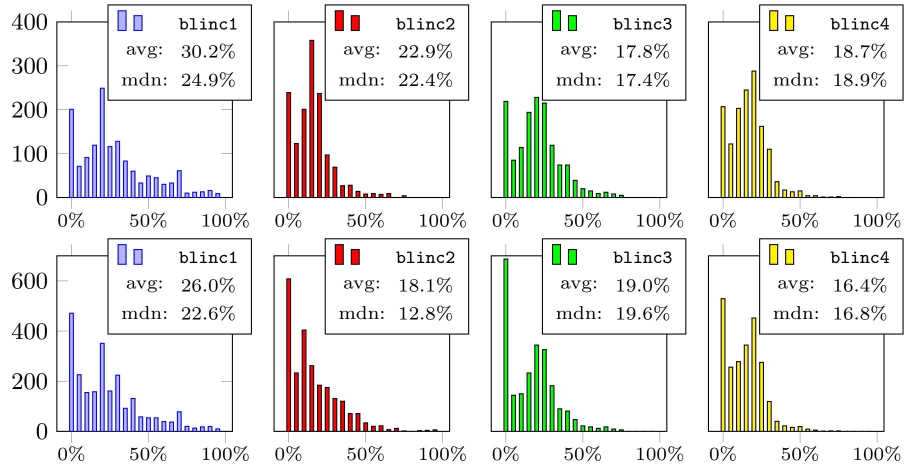

Fig. 5. Distribution of UEQ (top) and PEQ (bottom) benchmarks by ratio of skipped superpositions to all superpositions, showing also average (avg) and median (mdn). For example, using `blinc1`, on average 30.2%, resp. 26.0% of superpositions can be skipped in UEQ, resp. PEQ benchmarks.

More general irreducibility constraints were considered in completion [22] and in superposition [17], the latter using a semantic tree method for completeness. Ordering constraints [9,10,19] and unification constraints [8,27] have also been considered, usually moving them to the calculus level. Extending and generalizing our BLINC framework with such constraints is a future challenge.

8 Conclusions

We introduce reducibility constraints to block inferences during superposition reasoning. Our resulting BLINC calculus is refutationally complete and is extended with redundancy elimination, allowing us to maintain efficient reasoning when compared to state-of-the-art superposition proving. Integrating our approach with further inference-blocking constraints, such as blocking more general or outermost superpositions, is an interesting line for future work. Adapting our framework to domain-specific inference rules, e.g. in linear arithmetic or higher-order superposition, is another line for future work.

Other interesting directions are (i) the use of a stronger semantics of constraints, as in Definition 10 and (ii) a "hybrid calculus", improving on `blinc3`, where we still use constraints for blocking generating inferences but relax them whenever they prevent us from applying a simplification or a deletion rule.

Acknowledgements. We thank Konstantin Korovin for fruitful discussions. We acknowledge funding from the ERC Consolidator Grant ARTIST 101002685, the TU Wien SecInt Doctoral College, the FWF SFB project SpyCoDe F8504, and the WWTF ICT22-007 grant ForSmart.

References

1. Avenhaus, J., Hillenbrand, T., Löchner, B.: On Using Ground Joinable Equations in Equational Theorem Proving. Journal of Symbolic Computation **36**(1), 217–233 (2003). https://doi.org/10.1016/S0747-7171(03)00024-5
2. Baader, F., Nipkow, T.: Term Rewriting and All That. Cambridge University Press, USA (1998)
3. Bachmair, L.: Canonical Equational Proofs. Progress in theoretical computer science, Birkhäuser (1991)
4. Bachmair, L., Dershowitz, N.: Critical Pair Criteria for Completion. Journal of Symbolic Computation **6**(1), 1–18 (1988). https://doi.org/10.1016/S0747-7171(88)80018-X
5. Bachmair, L., Ganzinger, H.: Equational Reasoning in Saturation-Based Theorem Proving, vol. I, chap. 11, pp. 353–397 (1998)
6. Bachmair, L., Ganzinger, H.: Resolution Theorem Proving. In: Handbook of Automated Reasoning (in 2 volumes), pp. 19–99. Elsevier and MIT Press (2001)
7. Bachmair, L., Ganzinger, H., Lynch, C., Snyder, W.: Basic Paramodulation and Superposition. In: CADE. pp. 462–476 (1992)
8. Bhayat, A., Schoisswohl, J., Rawson, M.: Superposition with delayed unification. In: CADE. pp. 23–40 (2023)
9. Comon, H.: Solving Symbolic Ordering Constraints. International Journal of Foundations of Computer Science **01**(04), 387–411 (1990). https://doi.org/10.1142/S0129054190000278
10. Comon, H., Nieuwenhuis, R., Rubio, A.: Orderings, AC-Theories and Symbolic Constraint Solving (Extended Abstract). In: Annual IEEE Symposium on Logic in Computer Science. pp. 375–385 (1995). https://doi.org/10.1109/LICS.1995.523272
11. Denzinger, J., Kronenburg, M., Schulz, S.: DISCOUNT - A Distributed and Learning Equational Prover. J. Autom. Reason. **18**(2), 189–198 (1997). https://doi.org/10.1023/A:1005879229581
12. Dershowitz, N., Manna, Z.: Proving Termination with Multiset Orderings. Commun. ACM **22**(8), 465–476 (aug 1979). https://doi.org/10.1145/359138.359142
13. Duarte, A., Korovin, K.: Ground Joinability and Connectedness in the Superposition Calculus. In: IJCAR. pp. 169–187 (2022)
14. Gleiss, B., Kovács, L., Rath, J.: Subsumption Demodulation in First-Order Theorem Proving. In: IJCAR. pp. 297–315 (2020)
15. Hullot, J.M.: Canonical Forms and Unification. In: CADE. pp. 318–334 (1980)
16. Kapur, D., Musser, D.R., Narendran, P.: Only Prime Superpositions Need be Considered in the Knuth-Bendix Completion Procedure. Journal of Symbolic Computation **6**(1), 19–36 (1988). https://doi.org/10.1016/S0747-7171(88)80019-1
17. Kirchner, C., Kirchner, H., Rusinowitch, M.: Deduction with Symbolic Constraints. Research Report RR-1358, INRIA (1990)
18. Knuth, D.E., Bendix, P.B.: Simple Word Problems in Universal Algebras, pp. 342–376 (1983). https://doi.org/10.1007/978-3-642-81955-1_23
19. Korovin, K., Voronkov, A.: Knuth-Bendix Constraint Solving Is NP-Complete. In: Automata, Languages and Programming. pp. 979–992 (2001)
20. Kovács, L., Voronkov, A.: First-Order Theorem Proving and Vampire. In: CAV. pp. 1–35 (2013)
21. Löchner, B.: A Redundancy Criterion Based on Ground Reducibility by Ordered Rewriting. In: IJCAR. pp. 45–59 (2004)

22. Lynch, C., Snyder, W.: Redundancy Criteria for Constrained Completion. In: RTA. pp. 2–16 (1993)
23. McCune, W.: Solution of the Robbins Problem. Journal of Automated Reasoning **19**, 263–276 (1997)
24. Nieuwenhuis, R., Rubio, A.: Paramodulation-Based Theorem Proving. In: Handbook of Automated Reasoning, vol. I, chap. 7, pp. 371–443 (2001)
25. Nieuwenhuis, R., Rubio, A.: Basic Superposition is Complete. In: ESOP. pp. 371–389 (1992)
26. Nutt, W., Réty, P., Smolka, G.: Basic Narrowing Revisited. Journal of Symbolic Computation **7**(3-4), 295–317 (1989)
27. Reger, G., Suda, M., Voronkov, A.: Unification with Abstraction and Theory Instantiation in Saturation-Based Reasoning. In: TACAS. pp. 3–22
28. Schulz, S., Cruanes, S., Vukmirović, P.: Faster, Higher, Stronger: E 2.3. In: CADE. pp. 495–507 (2019)
29. Sutcliffe, G.: The Logic Languages of the TPTP World. Logic Journal of the IGPL (2022). https://doi.org/10.1093/jigpal/jzac068
30. Voronkov, A.: AVATAR: The Architecture for First-Order Theorem Provers. In: CAV. pp. 696–710 (2014). https://doi.org/10.1007/978-3-319-08867-9_46
31. Weidenbach, C., Dimova, D., Fietzke, A., Kumar, R., Suda, M., Wischnewski, P.: SPASS Version 3.5. In: CADE. pp. 140–145 (2009)
32. Zhang, H., Kapur, D.: Consider only General Superpositions in Completion Procedures. In: RTA. pp. 513–527 (1989)
33. Zhang, H., Kapur, D.: Unnecessary Inferences in Associative-Commutative Completion Procedures. In: Mathematical Systems Theory (1990). https://doi.org/10.1007/BF02090774

Appendix

Proofs

Lemma 5. *If $R_{\mathbb{C}} \models \mathbb{C}$, then for all $\mathbb{D} \succ \mathbb{C}$ we have $R_{\mathbb{D}} \models \mathbb{C}$.*

Proof. If a positive literal $s \simeq t$ of $\mathbb{C}$ is true in $R_{\prec\mathbb{C}}$, then $s \downarrow_{R_{\prec\mathbb{C}}} t$. Since no rules are ever removed during the model construction, then $s \downarrow_{R_{\mathbb{C}}} t$, $s \downarrow_{R_{\prec\mathbb{D}}} t$ and $s \downarrow_{R_\infty} t$. If a negative literal $s \not\simeq t$ of $\mathbb{C}$ is true in $R_{\prec\mathbb{C}}$, then $s \not\downarrow_{R_{\prec\mathbb{C}}} t$. Consider a productive closure $\mathbb{D} \succ \mathbb{C}$, producing a rule $l \to r$. By the productive closure definition in Definition 6, l is the maximal term in $\mathbb{D}$. If $l \to r$ reduces a subterm of $s \not\simeq t$, it means that $\mathbb{D} \prec \mathbb{C}$, contradiction. This also implies $s \not\downarrow_{R_\infty} t$. ☐

Lemma 6. *If $\mathcal{C} \cdot \theta = (l \simeq r \vee C' \mid \Gamma) \cdot \theta$ is productive, then $R_{\prec\mathcal{D}\cdot\sigma} \not\models C'\theta$ for any $\mathcal{D} \cdot \sigma \succ \mathcal{C} \cdot \theta$, and $R_\infty \not\models C'\theta$.*

Proof. All literals in $C'\theta$ are false in $R_{\prec\mathcal{C}\cdot\theta}$ by Definition 6. For all negative literals $(s \not\simeq t)\theta$ in $C'\theta$, if they are false then $s\theta \downarrow_{R_{\prec\mathcal{C}\cdot\theta}} t\theta$. Since no rules are ever removed during the model construction then $s\theta \downarrow_{R_{\prec\mathcal{D}\cdot\sigma}} t\theta$ and $s\theta \downarrow_{R_\infty} t\theta$. For all positive literals $(s \simeq t)\theta$ in $C'\theta$, if they are false in $R_{\prec\mathcal{C}\cdot\theta}$ then $s\theta \not\downarrow_{R_{\prec\mathcal{C}\cdot\theta}} t\theta$. By Definition 6, we have $l\theta \succeq s\theta$ and $l\theta \succeq t\theta$. Any closure $\mathcal{D} \cdot \sigma$ that produces a rule $l'\sigma \to r'\sigma$ which reduces $s\theta$ or $t\theta$ must have either have $l'\sigma \prec l\theta$, in which case either $\mathcal{D} \cdot \sigma \prec \mathcal{C} \cdot \theta$, or $l'\sigma = l\theta$, in which case whichever clause is bigger would not be productive due to Definition 6. ☐

Lemma 7. *R_∞ is non-overlapping, terminating and hence canonical.*

Proof. R_∞ is terminating since $l \succ r$ by construction for all rules $l \to r \in R_\infty$. Assume R_∞ is overlapping, meaning there are two different rules $l \to r$ and $l' \to r'$ such that w.l.o.g. $l \trianglelefteq l'$. This means that when generating the rule $l' \to r'$, l' is reducible by $l \to r$ (since it was generated by a smaller closure). But then $l' \to r'$ is not generated by definition of productive closure. Hence, R_∞ is non-overlapping. ☐

Lemma 8. *Let l be a term and C and D two clauses. If $C \succ D$ and $s \not\succ l$ for all terms in C, then $s \not\succ l$ for all terms in D as well.*

Proof. It follows from the bag extension definition of $\succ$ for terms. ☐

Lemma 1. (Unblocking Inferences) *Suppose $\mathcal{C}, \mathcal{D} \in N$ and θ and σ are substitutions irreducible in $R_{\prec\mathcal{C}\cdot\theta}$ and in $R_{\prec\mathcal{D}\cdot\sigma}$, respectively. If $\mathcal{C} \cdot \theta \succ \mathcal{D} \cdot \sigma$, $\Gamma\theta \supseteq \Delta\sigma$ and $\mathcal{C} \cdot \theta$ is unblocked w.r.t. N, then $\mathcal{D} \cdot \sigma$ is unblocked w.r.t. N.*

Proof. Let us assume contrary to the claim that the above conditions hold but $\mathcal{D} \cdot \sigma$ is blocked w.r.t. N. We consider the following two cases.

(i) There is an irreducibility constraint $\downarrow l \in \Delta\sigma$ which is active in $\mathcal{D} \cdot \sigma$ and not satisfied by $R_{\prec\mathcal{D}\cdot\sigma}$. By Lemma 8 and $\mathcal{C} \cdot \theta \succ \mathcal{D} \cdot \sigma$, it is also active in $\mathcal{C} \cdot \theta$. Moreover, we have that l is reducible in $R_{\prec\mathcal{D}\cdot\sigma}$, which implies by $R_{\prec\mathcal{D}\cdot\sigma} \subseteq R_{\prec\mathcal{C}\cdot\theta}$ that l is also reducible in $R_{\prec\mathcal{C}\cdot\theta}$. Hence, $\mathcal{C} \cdot \theta$ is blocked w.r.t. N, contradiction.

(ii) There is a one-step reducibility constraint $l \rightsquigarrow r \in \Gamma$ which is active in $\mathcal{D} \cdot \sigma$ and not satisfied by $R_{\prec \mathcal{D} \cdot \sigma}$. By Lemma 8 and $\mathcal{C} \cdot \theta \succ \mathcal{D} \cdot \sigma$, it is also active in $\mathcal{C} \cdot \theta$. Moreover, we have $l \to r \notin R_{\prec \mathcal{D} \cdot \sigma}$. If $l \to r \notin R_{\prec \mathcal{C} \cdot \theta}$, the constraint is not satisfied by $R_{\prec \mathcal{C} \cdot \theta}$ either, hence $\mathcal{C} \cdot \theta$ is blocked w.r.t. N, contradiction. Otherwise, there is a closure $\mathcal{C}' \cdot \sigma$ which produced $l \to r$ s.t. $\mathcal{C} \cdot \theta \succ \mathcal{C}' \cdot \rho \succeq \mathcal{D} \cdot \sigma$ and by definition of productive clause, the maximal term in $\mathcal{C}'\sigma$ is l. But then $s \succ l$ cannot be the case for any s occurring in $\mathcal{D}\sigma$, so the constraint is not active in $\mathcal{D}\sigma$, contradiction. ◻

Lemma 2. (Redundancy with Blocked Clauses) *Let $\mathcal{C}$ be a constrained clause. If $\mathcal{C}$ is blocked, then all ground instances of $\mathcal{C}$ are blocked w.r.t. N.*

Proof. Let $\mathcal{C} = C \mid \Gamma$ be a blocked constrained clause and assume to the contrary that $\mathcal{C} \cdot \theta$ is unblocked w.r.t. some constrained clause set N. The first case is when Γ contains two constraints $l \rightsquigarrow r$ and $l \rightsquigarrow r'$ active in $\mathcal{C}$ such that r and r' are not unifiable. Then, $s \succ l$ for some s in C implies $s\theta \succ l\theta$ for some $s\theta$ occurring in $C\theta$, and there are $l\theta \to r\theta, l\theta \to r'\theta \in R_{\prec \mathcal{C} \cdot \theta}$ since $\mathcal{C} \cdot \theta$ is unblocked w.r.t. N. Since r and r' are not unifiable, we get $r\theta \neq r'\theta$, so one of the rules is reducible by the other, which contradicts this rule being produced by Definition 6. Otherwise, Γ contains two constraints $l \rightsquigarrow r$ and $\Downarrow l$ active in $\mathcal{C}$. Again, $s \succ l$ for some s in C implies $s\theta \succ l\theta$ and $s\theta$ is in $C\theta$. We also have that $l\theta \to r\theta \in R_{\prec \mathcal{C} \cdot \theta}$ and $l\theta$ is also irreducible in $R_{\prec \mathcal{C} \cdot \theta}$, which is a contradiction. ◻

Lemma 3 (Unblocked $\mathsf{Sup}_{\Supset}$). *Suppose that (a) $\mathcal{D} = \underline{s \bowtie t} \vee D \mid \Gamma$ is a constrained clause in N, (b) $\mathcal{D} \cdot \theta$ a ground closure unblocked w.r.t. N, (c) θ is irreducible in $R_{\prec \mathcal{D} \cdot \theta}$, (d) $s\theta \succeq t\theta$, (e) $s\theta$ is reducible in $R_{\prec \mathcal{D} \cdot \theta}$.*

Then there exist a constrained clause $(\underline{l \simeq r} \vee C \mid \Pi) \in N$, a $\mathsf{Sup}_{\Supset}$-inference

$$(\mathsf{Sup}_{\Supset})\ \frac{\underline{l \simeq r} \vee C \mid \Pi \quad \underline{s[u] \bowtie t} \vee D \mid \Gamma}{(s[r] \bowtie t \vee C \vee D)\sigma \mid \Delta}$$

and a substitution τ such that (i) $\mathcal{D}\sigma\tau = \mathcal{D}\theta$, (ii) $\underline{l \simeq r} \vee C \mid \Pi \cdot \sigma\tau$ is productive, and $(s[r] \bowtie t \vee C \vee D)\sigma \mid \Delta \cdot \sigma\tau$ is unblocked w.r.t. N.

Proof. Consider the following set:

$$\Lambda := \left\{ l'\theta \mid l' \trianglelefteq s \text{ non-variable and } l'\theta \to r\theta \in R_{\prec \mathcal{D} \cdot \theta} \right\}$$

The set Λ cannot be empty, as it would contradict either that $\mathcal{D} \cdot \theta$ is unblocked w.r.t. N and θ irreducible in $R_{\prec \mathcal{D} \cdot \theta}$, or that $s\theta$ is reducible in $R_{\prec \mathcal{D} \cdot \theta}$. Hence, Λ is non-empty, and by well-foundedness of $\Supset$, there is (at least) one smallest element $l'\theta$ in Λ w.r.t. $\Supset$, corresponding to some rule $l'\theta \to r\theta \in R_{\prec \mathcal{D} \cdot \theta}$. There is a clause $(\underline{l \simeq r} \vee C \mid \Pi) \in N$ such that the ground closure $(\underline{l \simeq r} \vee C \mid \Pi) \cdot \theta$ is productive and produces $l'\theta \to r\theta$. Moreover, there is a $\mathsf{Sup}_{\Supset}$ inference

$$\frac{\underline{l \simeq r} \vee C \mid \Pi \quad \underline{s[l'] \bowtie t} \vee D \mid \Gamma}{(s[r] \not\simeq t \vee C \vee D)\sigma \mid \Delta}$$

where $\sigma = \mathsf{mgu}(l', l)$ and $\Delta = \Gamma\sigma \cup \mathcal{B}_{\Supset}(s\sigma, l\sigma) \cup \{l\sigma \rightsquigarrow r\sigma\}$. Let us denote the clause $(s[r] \not\simeq t \vee C \vee D)\sigma \mid \Delta$ by $\mathcal{D}'$. Take the instance $\mathcal{D}' \cdot \rho = ((s[r] \not\simeq t \vee C \vee D)\sigma \mid \Delta) \cdot \rho$ where $\theta = \sigma\rho$. By idempotence of σ we have $\mathcal{D}' \cdot \rho = \mathcal{D}' \cdot \theta$. Now assume contrary to the claim that $\mathcal{D}' \cdot \theta$ is blocked w.r.t. N, that is, either there is a variable x s.t. $x\sigma\theta$ is reducible in $R_{\prec \mathcal{D}' \cdot \theta}$, or $R_{\prec \mathcal{D}' \cdot \theta}$ violates some constraint in $\Delta\theta$ that is active in $\mathcal{D}' \cdot \theta$. In the first case, $x\theta$ is reducible, contradicting that both premises are unblocked w.r.t. N (since x is in one of the premises). We note here that $(\underline{l \simeq r} \vee C \mid \Pi) \cdot \theta$ cannot contain any reducible terms at or below variable positions either, otherwise it would follow from a smaller closure in N^* and would not be productive. Otherwise, $R_{\prec \mathcal{D}' \cdot \theta}$ violates some constraint in $\Delta\theta$ that is active in $\mathcal{D}' \cdot \theta$. We distinguish the following three cases:

(i) $R_{\prec \mathcal{D}' \cdot \theta}$ violates a constraint in $\Gamma\sigma\theta$. This case is similar to Lemma 1.

(ii) If $R_{\prec \mathcal{D}' \cdot \theta}$ violates a constraint in $\mathcal{B}_{\Supset}(s\sigma, l\sigma)\theta$, then by definition of $\mathcal{B}_{\Supset}$, there is a non-variable $u\sigma$ in $s\sigma$, s.t. $l\sigma \Supset u\sigma$ and $\downarrow u\sigma\theta \in \mathcal{B}_{\Supset}(s\sigma, l\sigma)\theta$ s.t. $u\sigma\theta$ is reducible in $R_{\prec \mathcal{D}' \cdot \theta}$. If u is a variable, we get that $u\sigma\theta$ is reducible in $R_{\prec \mathcal{D}' \cdot \theta}$, and $\mathcal{D} \cdot \theta$ is blocked w.r.t. N, contradiction. It is straightforward to prove that $\mathcal{D} \cdot \theta \succ \mathcal{D}' \cdot \theta$ (see, for example, Subcases 4.2 and 5.2 in Theorem 1). By $\mathcal{D} \cdot \theta \succ \mathcal{D}' \cdot \theta$, we have $R_{\prec \mathcal{D} \cdot \theta} \supseteq R_{\prec \mathcal{D}' \cdot \theta}$, hence $u\sigma\theta$ is also reducible in $R_{\prec \mathcal{D} \cdot \theta}$. Hence, due to u not being a variable, $u\theta \rightsquigarrow w\theta \in \Lambda$ for some w. But then, by stability of $\Supset$ under substitutions, $l\sigma \Supset u\sigma$ implies $l\sigma\theta \Supset u\sigma\theta$ implies $l\theta \Supset u\theta$, hence $l'\theta$ is not smallest w.r.t. $\Supset$ in Λ, contradiction.

(iii) If $R_{\prec \mathcal{D}' \cdot \theta}$ violates $\{l\sigma \rightsquigarrow r\sigma\}\theta$, then it violates $l\sigma\theta \rightsquigarrow r\sigma\theta$, which means $l\sigma\theta \to r\sigma\theta \notin R_{\prec \mathcal{D}' \cdot \theta}$. By construction of Λ, we have $l\sigma\theta \to r\sigma\theta = l\theta \to r\theta \in R_{\prec \mathcal{D} \cdot \theta}$, and as noted earlier, this rule is produced by $(\underline{l \simeq r} \vee D') \cdot \theta$. Moreover, we have $(\underline{l \simeq r} \vee D') \cdot \theta \succ \mathcal{D}' \cdot \theta$. But then, by definition of productive clause, $l\theta$ is the maximal term in $(\underline{l \simeq r} \vee D') \cdot \theta$, which means $s \succ l\theta$ cannot be the case for any term s occurring in $\mathcal{D}'\theta$, hence $l \rightsquigarrow r$ is not active in $\mathcal{D}'\theta$, contradiction.

We have covered all cases, which proves the claim. □

Theorem 1 (Model of $\mathcal{U}(N)$). *Let N be saturated up to redundancy and $\square \notin N$. Then for each $\mathbb{C} \in \mathcal{U}(N)$ we have $R_{\mathbb{C}} \models \mathbb{C}$.*

Proof. Assume, by contradiction, that $\square \notin N$ and $\mathcal{U}(N)$ contains a ground closure $\mathbb{C}$ such that $R_{\mathbb{C}} \not\models \mathbb{C}$. Since $\succ$ is well-founded, then N^* contains a minimal non-blocked closure $\mathcal{C} \cdot \theta$ such that $R_{\mathcal{C} \cdot \theta} \not\models \mathcal{C} \cdot \theta$. By induction hypothesis all closures $\mathcal{D} \cdot \sigma \in \mathcal{U}(N)$ such that $\mathcal{D} \cdot \sigma \prec \mathcal{C} \cdot \theta$ have $R_{\mathcal{D} \cdot \sigma} \models \mathcal{D} \cdot \sigma$, then we have $R_{\prec \mathcal{C} \cdot \theta} \models \mathcal{D} \cdot \sigma$ (and $R_{\mathcal{C} \cdot \theta} \models \mathcal{D} \cdot \sigma$). Consider the following cases.

Case 1. $\mathcal{C}$ is redundant w.r.t. N.

Proof. By definition, $\mathcal{C} \cdot \theta$ follows from smaller closures in $\mathcal{U}(N)$. But if $\mathcal{C} \cdot \theta$ is the minimal closure which is false in $R_{\mathcal{C} \cdot \theta}$, then all smaller $\mathcal{D} \cdot \sigma \in \mathcal{U}(N)$ are true in $R_{\mathcal{D} \cdot \sigma}$, which (as noted above) means that all smaller $\mathcal{D} \cdot \sigma \in \mathcal{U}(N)$ are true in $R_{\mathcal{C} \cdot \theta}$, which means $\mathcal{C} \cdot \theta$ is true in $R_{\mathcal{C} \cdot \theta}$, contradiction. □

Case 2. $\mathcal{C}$ contains a variable x such that $x\theta$ is reducible in $R_{\prec \mathcal{C} \cdot \theta}$.

Proof. Then, the reduced closure $\mathcal{C}\cdot\theta'$ of $\mathcal{C}\cdot\theta$ is unblocked w.r.t. N and $\mathcal{C}\cdot\theta' \neq \mathcal{C}\cdot\theta$. Then $\mathcal{C}\cdot\theta \succ \mathcal{C}\cdot\theta'$, and therefore $\mathcal{C}\cdot\theta'$ is true in $R_{\mathcal{C}\cdot\theta}$. Since $x\theta \downarrow_{R_{\mathcal{C}\cdot\theta}} t$, then $\mathcal{C}\cdot\theta$ is also true in $R_{\mathcal{C}\cdot\theta}$, contradiction. □

Case 3. There is a reductive inference $\mathcal{C}_1,\dots,\mathcal{C}_n \vdash \mathcal{D}$ with $\mathcal{C}_1,\dots,\mathcal{C}_n \in N$ which is redundant w.r.t. N such that (a) $\{\mathcal{C}_1\cdot\theta,\dots,\mathcal{C}_n\cdot\theta\} \subseteq \mathcal{U}(N)$, (b) $\mathcal{D}\cdot\theta$ is unblocked w.r.t. N, (c) $\mathcal{C}\cdot\theta = \max\{\mathcal{C}_1\cdot\theta,\dots,\mathcal{C}_n\cdot\theta\}$, and (d) $\mathcal{D}\cdot\theta \models \mathcal{C}\cdot\theta$.

Proof. Then $\mathcal{D}\cdot\theta$ is implied by closures in $\mathcal{U}(N)$ smaller than $\mathcal{C}\cdot\theta$ (since $\mathcal{C}\cdot\theta \in \mathcal{U}(N)$, see Definition 7). But since those closures are true in $R_{\mathcal{C}\cdot\theta}$, then $\mathcal{D}\cdot\theta$ is true, and since $\mathcal{D}\cdot\theta$ implies $\mathcal{C}\cdot\theta$, then $\mathcal{C}\cdot\theta$ is true in $R_{\mathcal{C}\cdot\theta}$, contradiction. □

Case 4. None of the previous cases apply, and $\mathcal{C}$ contains a *negative* literal which is selected in the clause, i.e., $\mathcal{C}\cdot\theta = (s \not\simeq t \vee C \mid \Gamma)\cdot\theta$ with $s \not\simeq t$ selected in $\mathcal{C}$.

Proof. Then either $s\theta \not\downarrow_{R_{\mathcal{C}\cdot\theta}} t\theta$ and $\mathcal{C}\cdot\theta$ is true and we are done or else $s\theta \downarrow_{R_{\mathcal{C}\cdot\theta}} t\theta$. By Definition 6(i) and (iv), $\mathcal{C}\cdot\theta$ is only productive if $(s \not\simeq t)\cdot\theta$ is false in $R_{\prec\mathcal{C}\cdot\theta}$ and $R_{\mathcal{C}\cdot\theta}$, so $s\theta \downarrow_{R_{\prec\mathcal{C}\cdot\theta}} t\theta$ iff $s\theta \downarrow_{R_{\mathcal{C}\cdot\theta}} t\theta$. W.l.o.g., let us assume $s\theta \succeq t\theta$.

Subcase 4.1. $s\theta = t\theta$.

Proof. Then s and t are unifiable and there is an equality resolution inference

$$\frac{\underline{s \not\simeq t} \vee C \mid \Gamma}{C\sigma \mid \Gamma\sigma}\ \sigma = \mathsf{mgu}(s,t)$$

with premise in N. Take the instance $\mathcal{D}\cdot\rho = (C\sigma \mid \Gamma\sigma)\cdot\rho$ of the conclusion such that $\sigma\rho = \theta$; it always exists since $\sigma = \mathsf{mgu}(s,t)$. Also, since the mgu is idempotent [2] then $\sigma\theta = \sigma(\sigma\rho) = \sigma\rho = \theta$, so $C\sigma\cdot\rho = C\sigma\cdot\theta$ and $\Gamma\sigma\theta = \Gamma\theta$. Showing that $\mathcal{C}\cdot\theta \succ \mathcal{D}\cdot\theta$ follows from the multiset of $C\sigma\theta$ being a strict submultiset of $C\theta$. If $C\sigma\cdot\rho$ is true in $R_{\prec\mathcal{C}\cdot\theta}$ then $(s \not\simeq t \vee C)\cdot\sigma\rho$ must also be true. Recall that Case 3 does not apply. But we have shown that this inference is reductive, with $\mathcal{C} \in N$, $\mathcal{C}\cdot\theta$ trivially maximal in $\{\mathcal{C}\cdot\theta\}$, $\mathcal{C}\cdot\theta \in \mathcal{U}(N)$ and that $\mathcal{D}\cdot\theta$ implies $\mathcal{C}\cdot\theta$. Moreover, by $\mathcal{C}\cdot\theta \succ \mathcal{D}\cdot\theta$ and $\Gamma\sigma\rho = \Gamma\sigma\theta = \Gamma\theta$, Lemma 1 applies and $\mathcal{D}\cdot\theta$ is also unblocked w.r.t. N. Also, using the contraposition of Lemma 2, we get that the conclusion is not blocked, and condition (2) of $\mathsf{EqRes}_{\Supset}$ is satisfied. So for Case 3 not to apply the inference must be non-redundant. Also since Case 1 does not apply then the premise is not redundant. This means that the set is not saturated, which is a contradiction. □

Subcase 4.2. $s\theta \succ t\theta$

Proof. Then (recall that $s\theta \downarrow_{R_{\prec\mathcal{C}\cdot\theta}} t\theta$) $s\theta$ must be reducible by some rule in $R_{\prec\mathcal{C}\cdot\theta}$. Let us say that this rule is $l\theta \to r\theta$, produced by a closure $\mathcal{D}\cdot\theta \in \mathcal{U}(N)$ smaller than $\mathcal{C}\cdot\theta$. Case 2 does not apply, hence $l\theta$ is not under a variable position in $s\theta$. The closure $\mathcal{D}\cdot\theta$ must be of the form $(l \simeq r \vee D)\cdot\theta$, with $l\theta \simeq r\theta$ strictly maximal in $\mathcal{D}\theta$, and $D\theta$ false in $R_{\mathcal{D}\cdot\theta}$. Also note that $\mathcal{D}$ cannot be redundant, or else $\mathcal{D}\cdot\theta$ would follow from smaller closures in $\mathcal{U}(N)$, but those closures (which are smaller than $\mathcal{D}\cdot\theta$ and therefore smaller than $\mathcal{C}\cdot\theta$) would be true, so $\mathcal{D}\cdot\theta$ would be also true in $R_{\prec\mathcal{D}\cdot\theta}$, so by Definition 6 it would not be productive. Then $l\theta = l'\theta$ for some subterm l' of s, meaning l is unifiable with l', meaning there exists a superposition inference

$$\frac{\underline{l \simeq r} \vee D \mid \Pi \quad \underline{s[l'] \not\simeq t} \vee C \mid \Gamma}{(s[r] \not\simeq t \vee C \vee D)\sigma \mid \Delta}$$

where $\sigma = \mathsf{mgu}(l, l')$ and $\Delta = \Gamma\sigma \cup \mathcal{B}_{\Supset}(s\sigma, l\sigma) \cup \{l\sigma \rightsquigarrow r\sigma\}$. Consider the instance $(s[r] \not\simeq t \vee C \vee D)\sigma \cdot \rho$ with $\sigma\rho = \theta$, and call this instance $\mathcal{C} \cdot \theta$. Showing that $\mathcal{C} \cdot \theta \succ \mathcal{C}' \cdot \theta$ amounts to showing $(s[l'] \not\simeq t \vee C)\theta \succ (s[r] \not\simeq t \vee C \vee D)\theta$, which means (after removing common elements from multisets), comparing $\{\{s[l']\theta, s[l']\theta, t\theta, t\theta\}\}$ with the multiset induced by $(s[r] \not\simeq t \vee D)\theta$. This follows from (i) $s[l']\theta \succ s[r]\theta$ due to $l\theta \succ r\theta$, and (ii) $\{\{s[l']\theta, s[l']\theta, t\theta, t\theta\}\}$ greater than the multiset induced by $L\theta$ for all $L\theta \in D\theta$ due to $s\theta \succeq l\theta \succ r\theta$ and $(l \simeq r)\theta$ being maximal in $\mathcal{D}\theta$.

Note also that $\mathcal{C} \cdot \theta$ is maximal in $\{\mathcal{C} \cdot \theta, \mathcal{D} \cdot \theta\}$ and by Lemma 3, we can select $\mathcal{D} \cdot \theta$ s.t. $\mathcal{C}' \cdot \theta$ is unblocked w.r.t. N. Since $\mathcal{C}' \cdot \theta$ is unblocked w.r.t. N, using the contraposition of Lemma 2, we get that the conclusion is not blocked, and condition (5) of the $\mathsf{Sup}_{\Supset}$ is satisfied. Also, since $D \cdot \theta$ is false in $R_{\prec \mathcal{C} \cdot \theta}$ (by Lemma 6) and $(s[r] \not\simeq t) \cdot \theta$ is false in $R_{\prec \mathcal{C} \cdot \theta}$ (since $(s \not\simeq t) \cdot \theta$ is in the false closure $\mathcal{C} \cdot \theta$, $l'\theta \downarrow_{R_{\prec \mathcal{C} \cdot \theta}} r\theta$, and the rewrite system is confluent), then in order for $\mathcal{C}' \cdot \theta$ to be true in $R_{\mathcal{C} \cdot \theta}$ it must be the case that $C\sigma\rho$ is true in $R_{\prec \mathcal{C} \cdot \theta}$. But if the latter is true then $\mathcal{C} \cdot \theta$ is true, in $R_{\prec \mathcal{C} \cdot \theta}$. In other words $\mathcal{C}' \cdot \theta$ implies $\mathcal{C} \cdot \theta$. Therefore again, since Case 1 and Case 3 do not apply, we conclude that the inference is non-redundant with non-redundant premises, so the set is not saturated, which is a contradiction. □

Case 5. None of the previous cases apply, so all selected literals in $\mathcal{C}$ are positive, i.e., $\mathcal{C} \cdot \theta = (s \simeq t \vee C \mid \Gamma) \cdot \theta$ with $s \simeq t$ selected in $\mathcal{C}$.

Proof. Then, since if the selection function does not select a negative literal then it must select all maximal ones, w.l.o.g. one (and only one) of the selected literals $s \simeq t$ maximal in $\mathcal{C}$ must have $s\theta \simeq t\theta$ maximal in $\mathcal{C}\theta$. Then if either $C \cdot \theta$ is true in $R_{\prec \mathcal{C} \cdot \theta}$, or $R_{\mathcal{C} \cdot \theta} = R_{\prec \mathcal{C} \cdot \theta} \cup \{s\theta \to t\theta\}$, or $s\theta = t\theta$, then $\mathcal{C} \cdot \theta$ is true in $R_{\mathcal{C} \cdot \theta}$ and we are done. Otherwise, $\mathcal{C} \cdot \theta$ is not productive, $C \cdot \theta$ is false in $R_{\prec \mathcal{C} \cdot \theta}$, and w.l.o.g. $s\theta \succ t\theta$.

Subcase 5.1. $s\theta \simeq t\theta$ maximal but not strictly maximal in $\mathcal{C}\theta$.

Proof. If this is the case, then there is at least one other maximal positive literal in the clause. Let $\mathcal{C} \cdot \theta = (s \simeq t \vee s' \simeq t' \vee C' \mid \Gamma) \cdot \theta$, where $s\theta = s'\theta$ and $t\theta = t'\theta$. Therefore s and s' are unifiable and there is an equality factoring inference:

$$\frac{\underline{s \simeq t} \vee \underline{s' \simeq t'} \vee C' \mid \Gamma}{(s \simeq t \vee t \not\simeq t' \vee C')\sigma \mid \Gamma\sigma} \; \sigma = \mathsf{mgu}(s, s')$$

Take the instance of the conclusion $\mathcal{C}' \cdot \rho = ((s \simeq t \vee t \not\simeq t' \vee C')\sigma \mid \Gamma\sigma) \cdot \rho$ with $\sigma\rho = \theta$. Showing that $\mathcal{C} \cdot \theta \succ \mathcal{C}' \cdot \rho$ and that $\mathcal{C}' \cdot \rho$ implies $\mathcal{C} \cdot \theta$ is straightforward based on [13]. Hence, the inference is reductive and from $\Gamma\sigma\rho = \Gamma\theta$, by Lemma 1, we have that $R_{\prec \mathcal{C}' \cdot \rho}$ satisfies $\Gamma\theta = \Gamma\sigma\rho$, hence $\mathcal{C}' \cdot \rho$ is unblocked w.r.t. N, which by Lemma 2 also means that the conclusion is not blocked, satisfying condition (3) of $\mathsf{EqFac}_{\Supset}$. Hence, we get a contradiction similarly to Subcase 4.1. □

Subcase 5.2. $s\theta \simeq t\theta$ strictly maximal in $\mathcal{C}\theta$, and $s\theta$ reducible (in $R_{\prec \mathcal{C} \cdot \theta}$).

Proof. This is similar to Subcase 4.2. If $s\theta$ is reducible, by Lemma 3, we have that there is a superposition which results in an unblocked closure $C \cdot \theta$ w.r.t. N. Let the reducing rule be $l\theta \to r\theta$, then (since $\mathcal{C} \cdot \theta$ is not productive) this is produced by some closure $\mathcal{D} \cdot \theta$ smaller than $\mathcal{C} \cdot \theta$, with $\mathcal{D} \cdot \theta = (l \simeq r \vee D) \cdot \theta$, with $l\theta \simeq r\theta$ maximal in $\mathcal{D}\theta$, $\mathcal{D}$ not redundant, and with $D \cdot \theta$ false in $R_{\prec \mathcal{D} \cdot \theta}$. Then there is a superposition inference

$$\frac{\underline{l \simeq r} \vee D \quad \underline{s[l'] \simeq t} \vee C \mid \Gamma}{(s[r] \simeq t \vee C \vee D)\sigma \mid \Delta}$$

where $\sigma = \mathsf{mgu}(l', l)$ and $\Delta = \Gamma\sigma \cup \mathcal{B}(s\sigma, l\sigma) \cup \{l\sigma \rightsquigarrow r\sigma\}$. Again taking the instance $\mathcal{C}' \cdot \rho = ((s[r] \simeq t \vee C \vee D)\sigma \mid \Gamma\sigma \cup \mathcal{B}(s\sigma, l\sigma)) \cdot \rho$ with $\sigma\rho = \theta$, we have $\mathcal{C}' \cdot \rho = \mathcal{C}' \cdot \theta$ and showing that $\mathcal{C}' \cdot \theta$ is smaller than $\mathcal{C} \cdot \theta$ is straightforward based on [13]. Furthermore since $D \cdot \theta$ and $C\theta$ are false in $R_{\prec \mathcal{C} \cdot \theta}$, then $\mathcal{C}' \cdot \theta$ is true in $R_{\prec \mathcal{C} \cdot \theta}$ iff $(s[r] \simeq t)\sigma \cdot \theta$ is. But since also $l'\theta \downarrow_{R_{\prec \mathcal{C} \cdot \theta}} r\theta$, then $(s[r] \simeq t)\sigma \cdot \theta$ implies $(s[l'] \simeq t)\sigma \cdot \theta$. So again by Lemma 3, $\mathcal{C}' \cdot \theta$ is unblocked w.r.t. N, and by Lemma 2 the conclusion is not blocked, satisfying condition (5) of $\mathsf{Sup}_{\Supset}$. Also, $\mathcal{C}' \cdot \theta$ implies $\mathcal{C} \cdot \theta$. Again this means we have a contradiction. □

Subcase 5.3. $s\theta \simeq t\theta$ strictly maximal in $\mathcal{C}\theta$, and $s\theta$ irreducible (in $R_{\prec \mathcal{C} \cdot \theta}$).

Proof. Since $\mathcal{C} \cdot \theta$ is not productive, and at the same time all criteria in Definition 6 except (iv) are satisfied, it must be that condition (iv) is not, that is $C\theta$ must be true in $R_{\mathcal{C} \cdot \theta} = R_{\prec \mathcal{C} \cdot \theta} \cup \{s\theta \to t\theta\}$. Then this must mean we can write $C\theta = (s' \simeq t' \vee C')\theta$, where the latter literal is the one that becomes true with the addition of $\{s\theta \to t\theta\}$, whereas without that rule it was false.

But this means that $s'\theta \downarrow_{R_{\prec \mathcal{C} \cdot \theta}} t'\theta$ such that any rewrite proof needs at least one step where $s\theta \to t\theta$ is used, since $s\theta$ is irreducible by $R_{\prec \mathcal{C} \cdot \theta}$. W.l.o.g. say $s'\theta \succ t'\theta$. Since $\{s\theta, t\theta\} \succ \{s'\theta, t'\theta\}$, $s\theta \succ t\theta$, and $s'\theta \succ t'\theta$, then $s\theta \succeq s'\theta \succ t'\theta$, which implies $t'\theta \not\succeq s\theta$, which implies $s\theta \to t\theta$ cannot be used to reduce $t'\theta$, and similarly, nor to reduce $s'\theta$ if $s\theta \succ s'\theta$. Thus the only way it can reduce $s'\theta$ or $t'\theta$ is if $s\theta = s'\theta$. This means there is an equality factoring inference:

$$\frac{\underline{s' \simeq t'} \vee \underline{s \simeq t} \vee C' \mid \Gamma}{(s' \simeq t' \vee t \not\simeq t' \vee C')\sigma \mid \Gamma\sigma,} \; \sigma = \mathsf{mgu}(s, s')$$

Taking $\theta = \sigma\rho$, we see that the instance of the conclusion $(s' \simeq t' \vee t \not\simeq t' \vee C')\sigma \cdot \rho$ is smaller than the instance of the premise $(s' \simeq t' \vee s \simeq t \vee C') \cdot \sigma\rho$ (see Subcase 5.1).

But we have said that $s'\theta \downarrow_{R_{\prec \mathcal{C} \cdot \theta}} t'\theta$, where the first rewrite step had to take place by rewriting $s'\theta = s\theta \to t\theta$, and the rest of the rewrite proof then had to use only rules from $R_{\prec \mathcal{C} \cdot \theta}$. In other words, this means $t\theta \downarrow_{R_{\prec \mathcal{C} \cdot \theta}} t'\theta$. As such, the literal $(t \not\simeq t') \cdot \theta$ is false in $R_{\prec \mathcal{C} \cdot \theta}$, and so the conclusion is true in $R_{\prec \mathcal{C} \cdot \theta}$ iff rest of the closure is true in $R_{\prec \mathcal{C} \cdot \theta}$. But if the rest of the closure $(s' \simeq t' \vee C')\sigma \cdot \rho$ is true then so is $\mathcal{C} \cdot \theta$, so that instance of the conclusion implies $\mathcal{C} \cdot \theta$. The inference being reductive and $\sigma\rho = \theta$, by Lemma 1 we also have that instance of the conclusion is unblocked w.r.t. N. Hence, by Lemma 2, condition (3) of $\mathsf{EqFac}_{\Supset}$ is satisfied. Once again, this leads to a contradiction since none of Cases 1 and 3 apply and therefore the set must not be saturated. □

Definition 11. Given a saturated set of clauses N and a constrained clause $\mathcal{C} \in N$. Let $\mathcal{C} \cdot \theta$ be a ground closure where θ is irreducible w.r.t. $R_{\prec \mathcal{C} \cdot \theta}$. We define the *blocked depth* of $\mathcal{C} \cdot \theta$ as:

$$bd(\mathcal{C} \cdot \theta) := \begin{cases} 0 & \text{if } \mathcal{C} \cdot \theta \text{ is unblocked w.r.t. } N, \\ 1 + \max_{1 \leq i \leq n}(bd(\mathcal{C}_i \cdot \theta)) & \text{if there is a } \mathsf{BLINC} \text{ inference} \\ & \mathcal{C}_1, ..., \mathcal{C}_n \vdash \mathcal{C} \text{ s.t. } \mathcal{C}_1, ..., \mathcal{C}_n \in N \end{cases}$$

Lemma 9. *Given a saturated set of clauses N, every closure $\mathcal{C} \cdot \theta \in N^*$ with irreducible θ follows from some closures $\mathcal{D}_1 \cdot \theta, ..., \mathcal{D}_n \cdot \theta \in \mathcal{U}(N)$.*

Proof. By induction on $bd(\mathcal{C} \cdot \theta)$. If $bd(\mathcal{C} \cdot \theta) = 0$, $\mathcal{C} \cdot \theta \in \mathcal{U}(N)$ trivially follows from itself. Otherwise, $\mathcal{C} \cdot \theta$ is blocked w.r.t. N but θ is irreducible in $R_{\prec \mathcal{C} \cdot \theta}$, so there must be a BLINC inference $\mathcal{C}_1, ..., \mathcal{C}_n \vdash \mathcal{C}$ with $\mathcal{C}_1, ..., \mathcal{C}_n \in N$. By definition of blocked depth, for all $\mathcal{C}_i$, $bd(\mathcal{C}_i \cdot \theta) < bd(\mathcal{C} \cdot \theta)$, hence by induction hypothesis, they are all implied by some closures $\mathcal{D}_{i1} \cdot \theta, ..., \mathcal{D}_{im_i} \cdot \theta \in \mathcal{U}(N)$ for $1 \leq i \leq n$. Since these inferences are sound, we have that $\mathcal{C} \cdot \theta$ also follows from $\mathcal{D}_{11} \cdot \theta, ..., \mathcal{D}_{nm_n} \cdot \theta \models \mathcal{C} \cdot \theta$. □

Theorem 2 (Model of N^*). *Let N be a saturated set of clauses. Every blocked closure $\mathcal{C} \cdot \theta \in N^*$ follows from $\mathcal{U}(N)$.*

Proof. By Lemma 9, we only have to consider $\mathcal{C} \cdot \theta$ with reducible θ in $R_{\prec \mathcal{C} \cdot \theta}$. In this case, we have $x_1, ..., x_n$ s.t. $x_i \theta = s_i \to^+_{R_\infty} t_i$ for all i and some normal form t_i. Let us consider $\theta' := (\theta \setminus \{x_i \mapsto s_i \mid i\}) \cup \{x_i \mapsto t_i \mid i\}$ which is now irreducible. From Lemma 9, we get that $\mathcal{C} \cdot \theta'$ follows from closures in $\mathcal{U}(N)$ and by $x_i \theta \downarrow_{R_\infty} t_i$ for each i, $\mathcal{C} \cdot \theta$ also follows from these closures in $\mathcal{U}(N)$. □

Lemma 10. *Let $\mathcal{C} = C \mid \Gamma$ and $\mathcal{D} = D \mid \Delta$ be well-behaved constrained clauses, and σ a substitution s.t. $\mathcal{C} \succ \mathcal{D}\sigma$ and $\Delta\sigma \subseteq \Gamma$. Each ground closure $\mathcal{C} \cdot \theta$ is blocked relative to the ground closure $\mathcal{D} \cdot \sigma\theta$.*

Proof. Take a substitution θ s.t. $\mathcal{C}\theta$ is ground. By $\mathcal{C} \succ \mathcal{D}\sigma$, then $\mathcal{D}\sigma\theta$ is also ground. Let $\mathcal{C} \cdot \theta'$ be the reduced closure of $\mathcal{C} \cdot \theta$, and let $\mathcal{D} \cdot \rho$ be the reduced closure of $\mathcal{D} \cdot \sigma\theta'$. We show that $\mathcal{C} \cdot \theta$ is blocked relative to $\mathcal{D} \cdot \rho$. Let N be a set of constrained clauses. Assume contrary to the claim that $\mathcal{C} \cdot \theta$ and hence also $\mathcal{C} \cdot \theta'$ is unblocked w.r.t. N, while $\mathcal{D} \cdot \rho$ and hence $\mathcal{D} \cdot \sigma\theta'$ are blocked w.r.t. N. Note that we have $\mathcal{C} \cdot \theta \succeq \mathcal{C} \cdot \theta' \succ \mathcal{D} \cdot \sigma\theta' \succeq \mathcal{D} \cdot \rho$. We consider two cases.

Case 1. There is $\downarrow l\rho \in \Delta\rho$ which is active in $\mathcal{D}\rho$ and $l\rho$ is reducible in $R_{\prec \mathcal{D} \cdot \rho}$.

Proof. Then, by $\Delta\sigma \subseteq \Gamma$, we have $\downarrow l'\theta' \in \Gamma\theta'$ s.t. $l' = l\sigma$ and either the constraint is not active in $\mathcal{C} \cdot \theta'$ or $l'\theta'$ is irreducible in $R_{\prec \mathcal{C} \cdot \theta'}$. If $l'\theta' = l\rho$, then by $\mathcal{C} \cdot \theta' \succ \mathcal{D} \cdot \rho$ and $R_{\prec \mathcal{C} \cdot \theta'} \supseteq R_{\prec \mathcal{D} \cdot \rho}$, we have that $\downarrow l'\theta'$ is active in $\mathcal{C} \cdot \theta'$ and reducible in $R_{\prec \mathcal{C} \cdot \theta'}$, hence $\mathcal{C} \cdot \theta'$ is blocked w.r.t. N, contradiction.

Otherwise, $l'\theta' \neq l\rho$ and there is $u' \trianglelefteq l'$ s.t. $u'\theta'$ is reducible in $R_{\prec \mathcal{D} \cdot \sigma\theta'}$. By well-behavedness of $\mathcal{C}$ and due to $\downarrow l' \in \Gamma$, in all cases we have $u'\theta' \in \Gamma\theta'$. Since $\downarrow l$ is active in $\mathcal{D}$ by well-behavedness of $\mathcal{D}$, we have $s \succ l\sigma = l' \succeq u'$ for some s in $\mathcal{D}\sigma$ and $\mathcal{C} \succ \mathcal{D}\sigma$, which implies $\downarrow u'\theta'$ is also active in $\mathcal{C} \cdot \theta'$ and $u'\theta'$ reducible in $R_{\prec \mathcal{C} \cdot \theta'}$, hence $\mathcal{C} \cdot \theta'$ is blocked w.r.t. N, contradiction.

Case 2. There is $l\rho \rightsquigarrow r\rho \in \Delta\rho$ such that it is active in $\mathcal{D}\rho$ and $l\rho \to r\rho \notin R_{\prec\mathcal{D}\cdot\rho}$.

Proof. Then by $\Delta\sigma \subseteq \Gamma$, we have $l'\theta' \rightsquigarrow r'\theta' \in \Gamma\theta'$ s.t. $l' = l\sigma$ and either the constraint is not active in $\mathcal{C}\cdot\theta'$ or $l'\theta' \to r'\theta' \in R_{\prec\mathcal{C}\cdot\theta'}$. If $l'\theta' = l\rho$, then by $\mathcal{C}\cdot\theta' \succ \mathcal{D}\cdot\rho$, we have that $l'\theta' \rightsquigarrow r'\theta'$ is active in $\mathcal{C}\cdot\theta'$. Then it must be that $l'\theta' \to r'\theta' \in R_{\prec\mathcal{C}\cdot\theta'}$. By the well-behavedness condition that l' contains all variables of r' and l contains all variables of r, we must have $r'\theta' = r\rho$ as well. Then, there is a closure $\mathcal{C}'\cdot\sigma$ which produced $l'\theta' \to r'\theta'$ s.t. $\mathcal{C}\cdot\theta' \succ \mathcal{C}'\cdot\eta \succeq \mathcal{D}\cdot\rho$ and by definition of productive clause, the maximal term in $\mathcal{C}'\eta$ is $l'\theta'$. But then $s \succ l'\theta'$ cannot be the case for any s occurring in $\mathcal{D}\rho$, so the constraint is not active in $\mathcal{D}\rho$, contradiction.

Otherwise, $l'\theta' \neq l\rho$ and there is $u' \trianglelefteq l'$ s.t. $u'\theta'$ is reducible in $R_{\prec\mathcal{D}\cdot\sigma\theta'}$. Note that $u' \neq l'$ as that would mean that l is a variable which by definition cannot happen (see Definition 1). So $u' \triangleleft l'$, and by well-behavedness of $\mathcal{C}$, we have $u'\theta' \in \Gamma\theta'$. Since $\downarrow l$ is active in $\mathcal{D}$ by well-behavedness of $\mathcal{D}$, we have $s \succ l\sigma = l' \succeq u'$ for some s in $\mathcal{D}\sigma$ and $\mathcal{C} \succ \mathcal{D}\sigma$, which implies $\downarrow u'\theta'$ is also active in $\mathcal{C}\cdot\theta'$ and $u'\theta'$ reducible in $R_{\prec\mathcal{C}\cdot\theta'}$, and again, $\mathcal{C}\cdot\theta'$ is blocked w.r.t. N, contradiction.

The above cases have shown that $\mathcal{C}\cdot\theta'$ is blocked relative to $\mathcal{D}\cdot\rho$ which implies that $\mathcal{C}\cdot\theta$ is also blocked relative to $\mathcal{D}\cdot\rho$. Finally, we argue that $\mathcal{C}\cdot\theta$ is blocked relative to $\mathcal{D}\cdot\sigma\theta$. If $\mathcal{D}\cdot\rho$ is the reduced closure of $\mathcal{D}\cdot\sigma\theta$, we get the claim by definition. Otherwise, there is some rule $l \to r \in R_{\prec\mathcal{C}\cdot\theta} \setminus R_{\prec\mathcal{D}\cdot\sigma\theta}$ which reduces $x\sigma\theta$ for some variable x in $\mathcal{D}$. This rule is produced by a closure greater than $\mathcal{D}\cdot\sigma\theta$, which means that $x\sigma\theta$ has to be the greatest term in $\mathcal{D}\cdot\theta$. By well-behavedness of $\mathcal{D}$, then it cannot be constrained, as that would mean $x \succ s$ for some non-variable term s in $\mathcal{D}$ (see Definition 1). Then, $\mathcal{D}\cdot\sigma\theta$ is unblocked w.r.t. N and $\mathcal{C}\cdot\theta$ is trivially blocked relative to it. □

Lemma 4. (Relatively Blocked Well-Behavedness) *Let $\mathcal{C} = C \mid \Gamma$ and $\mathcal{D} = D \mid \Delta$ be well-behaved constrained clauses, and σ a substitution. Then $\mathcal{C}$ is blocked relative to $\mathcal{D}$ if $\mathcal{C} \succ \mathcal{D}\sigma$ and $\Gamma \supseteq \Delta\sigma$.*

Proof. It follows from Lemma 10. □

Theorem 3. (BLINC Demodulation) $\mathsf{Dem}_{\Supset}$ *is a simplification rule. That is, $C[l\sigma] \mid \Gamma$ is redundant w.r.t. any constrained clause set that contains $l \simeq r \mid \Delta$ and $C[r\sigma] \mid \Gamma$.*

Proof. We present the rule again here for clarity:

$$(\mathsf{Dem}_{\Supset})\ \frac{l \simeq r \mid \Delta \quad \cancel{C[l\sigma] \mid \Gamma}}{C[r\sigma] \mid \Gamma} \qquad \text{where} \quad \begin{array}{l}(1)\ l\sigma \succ r\sigma,\\ (2)\ C[l\sigma] \succ (l \simeq r)\sigma,\\ (3)\ \Delta\sigma \subseteq \Gamma.\end{array}$$

Let $\mathcal{C} = C[l\sigma] \mid \Gamma$ and $\mathcal{D} = C[r\sigma] \mid \Gamma$ and let N be a set of constrained clauses s.t. $(l \simeq r \mid \Delta)$ and $\mathcal{D}$ are in N. Take a grounding substitution θ. If $\mathcal{C}\cdot\theta$ is blocked w.r.t. N, we are done. Otherwise, take the reduced closure $\mathcal{C}\cdot\theta'$ of $\mathcal{C}\cdot\theta$. By Lemma 10, $\mathcal{C}\cdot\theta'$ is blocked relative to the closure $(l \simeq r \mid \Delta)\cdot\sigma\theta'$ which is then also unblocked w.r.t. N and hence is in $\mathcal{U}(N)$. Moreover, this closure together with the instance $\mathcal{D} \cdot \theta'$ imply $\mathcal{C} \cdot \theta'$. Moreover, $\mathcal{C} \cdot \theta' \succ (l \simeq r \mid \Delta) \cdot \sigma\theta'$ follows from $\mathcal{C} \succ (l \simeq r)\sigma$ and $\mathcal{C} \cdot \theta' \succ \mathcal{D} \cdot \theta'$ follows from the monotonicity of $\succ$. Lemma 1 also applies due to $\mathcal{C} \cdot \theta' \succ \mathcal{D} \cdot \theta'$ and θ' being irreducible, hence $R_{\prec \mathcal{D} \cdot \theta'}$ satisfies $\Gamma\theta'$, and hence $\mathcal{D} \cdot \theta'$ is unblocked w.r.t. N and is in $\mathcal{U}(N)$ as well. Consequently, $\mathcal{C} \cdot \theta'$ and hence also $\mathcal{C} \cdot \theta$ follow from smaller closures in $\mathcal{U}(N)$. Hence $\mathcal{C}$ is redundant w.r.t. N. ☐

Theorem 4. (BLINC Subsumption) $\mathsf{Subs}_{\Supset}$ *is a deletion rule. That is, $C \mid \Gamma$ is redundant w.r.t. any constrained clause set that contains $D \mid \Delta$.*

Proof. We present the rule here again for clarity:

$$(\mathsf{Subs}_{\Supset})\ \frac{D \mid \Delta \quad \cancel{C \mid \Gamma}}{} \qquad \text{where} \quad \begin{array}{l}(1)\ D\sigma \subsetneq C,\\ (2)\ \Delta\sigma \subseteq \Gamma,\end{array} \ \text{for some substitution } \sigma.$$

Let $\mathcal{C} = C \mid \Gamma$ and $\mathcal{D} = D \mid \Delta$ and let N be a set of constrained clauses s.t. $\mathcal{D} \mid \Delta \in N$. Take a grounding substitution θ. If $\mathcal{C} \cdot \theta$ is blocked w.r.t. N, we are done. Otherwise, take the reduced closure $\mathcal{C} \cdot \theta'$ of $\mathcal{C} \cdot \theta$. By condition (1), we have $\mathcal{C}\theta' \succ \mathcal{D}\sigma\theta'$ and hence Lemma 10 implies $\mathcal{C} \cdot \theta'$ is blocked relative to the closure $\mathcal{D} \cdot \sigma\theta'$, so $\mathcal{D} \cdot \sigma\theta'$ is also unblocked w.r.t. N and is in $\mathcal{U}(N)$. Moreover, $\mathcal{D} \cdot \sigma\theta'$ then implies $\mathcal{C} \cdot \theta'$, hence $\mathcal{C} \cdot \theta'$ and also $\mathcal{C} \cdot \theta$ follow from smaller closures in $\mathcal{U}(N)$. Hence $\mathcal{C}$ is redundant w.r.t. N. ☐

Theorem 5. (BLINC Blocking) $\mathsf{Block}_{\Supset}$ *is a deletion rule. That is, $C \mid \Gamma$ is redundant w.r.t. any constrained clause that contains $l \simeq r \mid \Delta$.*

Proof. We present the rule here again for clarity:

$$(\mathsf{Block}_{\Supset})\ \frac{l \simeq r \mid \Delta \quad \cancel{C \mid \Gamma}}{} \quad \text{where} \quad \begin{array}{l}(1)\ C \succ (l \simeq r)\sigma \text{ and } l\sigma \succ r\sigma,\\ (2)\ \Delta\sigma \subseteq \Gamma,\\ (3)\ \text{either (i) } {\downarrow} l\sigma \in \Gamma\\ \quad\ \text{or (ii) } l\sigma \rightsquigarrow r' \in \Gamma \text{ and } r' \succ r\sigma.\end{array}$$

Let $\mathcal{C} = C \mid \Gamma$ and let N be a set of constrained clauses s.t. $(l \simeq r \mid \Delta) \in N$. Take a grounding substitution θ. If $\mathcal{C} \cdot \theta$ is blocked w.r.t. N, we are done. Otherwise, take the reduced closure $\mathcal{C} \cdot \theta'$ of $\mathcal{C} \cdot \theta$. By Lemma 4, $\mathcal{C} \cdot \theta'$ is blocked relative to the closure $(l \simeq r \mid \Delta) \cdot \sigma\theta'$, which is also unblocked w.r.t. N. We consider the following two cases:

Case 1. $l\sigma\theta'$ is irreducible in $R_{\prec (l \simeq r \mid \Delta) \cdot \sigma\theta'}$.

Proof. By $l\sigma \succ r\sigma$, $(l \simeq r \mid \Delta) \cdot \sigma\theta'$ is also productive, and by $C \succ (l \simeq r)\sigma$ we have $l\sigma\theta' \to r\sigma\theta' \in R_{\prec \mathcal{C} \cdot \theta'}$.

(i) If ${\downarrow} l\sigma \in \Gamma$, then ${\downarrow} l\sigma\theta' \in \Gamma\theta'$, and by $C \succ (l \simeq r)\sigma$, we have $C\theta' \succ (l \simeq r)\sigma\theta'$, which implies $s \succ l$ for some s in $C\theta'$ and ${\downarrow} l\sigma\theta'$ is active in $C\theta'$. Hence, $R_{\prec \mathcal{C} \cdot \theta'} \supseteq R_{\prec (l \simeq r \mid \Delta) \cdot \sigma\theta'}$ violates ${\downarrow} l\sigma\theta'$, so $\mathcal{C} \cdot \theta'$ is blocked w.r.t. N, contradiction.

(ii) If $l\sigma \rightsquigarrow r' \in \Gamma$ with $r' \succ r\sigma$, then $l\sigma\theta' \rightsquigarrow r'\theta' \in \Gamma\theta'$, and by $C \succ (l \simeq r)\sigma$, we have $C\theta' \succ (l \simeq r)\sigma\theta'$, which implies $s \succ l$ for some s in $C\theta'$ and $l\sigma\theta' \rightsquigarrow r'\theta'$ is active in $C\theta'$. Hence, by $r' \neq r\sigma$, $R_{\prec \mathcal{C}\cdot\theta'} \supseteq R_{\prec(l\simeq r|\Delta)\cdot\sigma\theta'}$ violates $l\sigma\theta' \rightsquigarrow r'\theta'$, so $\mathcal{C}\cdot\theta'$ is blocked w.r.t. N, contradiction.

Case 2. $l\sigma\theta'$ is reducible by some $u \rightarrow w \in R_{\prec(l\simeq r|\Delta)\cdot\sigma\theta'}$.

Proof. Then, there is some non-variable $u' \trianglelefteq l\sigma$ s.t. $u'\theta' = u$ (at or below a variable position contradicts θ' being irreducible). If $u' \neq l\sigma$, then by well-behavedness, since u' is non-variable, $\downarrow u' \in \Gamma$. Then, by $C\theta' \succ (l \simeq r)\sigma\theta'$, there is $s \succ l\sigma\theta' \succ u'\theta'$ for some s in $C\theta'$ and $\downarrow u'\theta'$ is active in $C\theta'$. Hence $R_{\prec\mathcal{C}\cdot\theta'} \supseteq R_{\prec(l\simeq r|\Delta)\cdot\sigma\theta'}$ violates $\Gamma\theta'$, contradiction. Otherwise, $u' = l\sigma$. If $\downarrow l\sigma \in \Gamma$, we get a contradiction similarly as in Case 1(i) above. If $l\sigma \rightsquigarrow r' \in \Gamma$ with $r' \succ r\sigma$, then $l\sigma\theta' \rightsquigarrow r'\theta' \in \Gamma\theta'$, and this constraint is active in $C\theta'$ for similar reasons as in Case 1(ii). Now, assume $l\sigma\theta' \rightarrow r'\theta' \in R_{\prec\mathcal{C}\cdot\theta'}$. This can only happen if $r'\theta' = w$. But then, by $r' \succ r\sigma$, we have $r'\theta' \succ r\sigma\theta'$, which implies that the closure producing $l\sigma\theta' \rightarrow r'\theta'$ is greater than $(l \simeq r)\cdot\sigma\theta'$, contradiction. $\square$

Theorem 6. (BLINC Orientation) $\mathsf{Orient}_{\Supset}$ *is a deletion rule. That is,* $C \mid \Gamma \cup \{l \rightsquigarrow r\}$ *is redundant w.r.t. any constrained clause set.*

Proof. We present the rule here again for clarity:

$$(\mathsf{Ord}_{\Supset})\ \frac{\cancel{C \mid \Gamma \cup \{l \rightsquigarrow r\}}}{} \quad \text{where} \quad \begin{array}{l}(1)\ r \succ l,\\ (2)\ C \succ (l \simeq r).\end{array}$$

Take a grounding substitution θ and let N be a set of constrained clauses. We argue that $(C \mid \Gamma)\cdot\theta$ is blocked w.r.t. N. We have $C\theta \succ (l \simeq r)\theta$ from $C \succ (l \simeq r)$, so in order for the instance to be unblocked w.r.t. N, it must be the case that $l\theta \rightarrow r\theta \in R_{\prec(C|\Gamma)\cdot\theta}$. This cannot be the case by Definition 6 and $r\theta \succ l\theta$, so the instance is blocked w.r.t. N. $\square$