



HAL
open science

Automata, Resolution, and Cut-elimination

Guillaume Burel, Gilles Dowek, Ying Jiang

► **To cite this version:**

Guillaume Burel, Gilles Dowek, Ying Jiang. Automata, Resolution, and Cut-elimination. 2016. hal-04063102v1

HAL Id: hal-04063102

<https://inria.hal.science/hal-04063102v1>

Preprint submitted on 13 Apr 2023 (v1), last revised 13 Apr 2023 (v2)

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

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

Automata, Resolution, and Cut-elimination

Guillaume Burel¹, Gilles Dowek², and Ying Jiang³

¹ ENSIIE/Samovar, 1 square de la Résistance, 91025 Évry Cedex, France,
`guillaume.burel@ensiie.fr`

² Inria, LSV, ENS-Cachan, 61 Avenue du Président Wilson, 94230 Cachan, France,
`gilles.dowek@inria.fr`

³ State Key Laboratory of Computer Science, Institute of Software, Chinese
Academy of Sciences, 100190 Beijing, China, `jy@ios.ac.cn`

Abstract. Automata, Resolution, and Cut-elimination are three methods to prove the decidability of provability in various logics. The goal of this paper is to understand the relation between these three methods, using Alternating pushdown systems as a unifying example. To this aim, we propose a new Resolution based saturation method for Polarized sequent calculus modulo theory and its associated cut-elimination theorem. We then show how it can be used to prove the decidability of Alternating pushdown systems.

1 Introduction

Cut-elimination and the construction of a finite-state automaton recognizing the provable propositions are two methods to prove the decidability of provability in inference systems. For instance, Kleene has proved the decidability of classical and constructive propositional logic by proving first a cut-elimination theorem for these logics and then the finiteness of the search space for a cut-free proof of a given proposition [14]. In contrast, A. Bouajjani, J. Esparza, and O. Maler, have proved the decidability of reachability in Alternating pushdown systems by constructing a finite-state automaton recognizing the inverse reachable configurations [3].

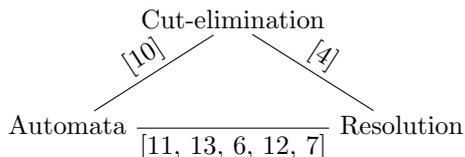
In previous work [10] we have shown that this methodological difference between cut-elimination and the construction of an automaton is not so strong as it looks: for each Alternating pushdown system, a Natural-Deduction style inference system can be defined, such that the inverse reachability in this Alternating pushdown system is reduced to the provability in the inference system, which in turn can be proved decidable, using a saturation algorithm and a cut-elimination theorem. However, the inference system, the saturation algorithm, and the cut-elimination theorem, presented in [10] are somehow *ad hoc*, as they are specific to Alternating pushdown systems. This raises the question of the possibility to prove this decidability result using already known inference systems, such as Polarized sequent calculus modulo theory [8, 9], and already known saturation algorithm and cut-elimination theorem.

A saturation algorithm for Polarized sequent calculus modulo theory has been presented in previous work [4]. This saturation algorithm transforms a rewrite

system into a set of clauses, saturates this set of clauses with Ordered resolution with selection and transforms back the saturated set of clauses into a rewrite system.

Interestingly enough, the decidability of Alternating pushdown systems has also been proved independently, using a saturation by Resolution method to transform Alternating pushdown systems into automata [11, 13, 6, 12, 7].

So, [10] relates cut-elimination and automata, [4] relates cut-elimination and Resolution, and [11, 13, 6, 12, 7] relate Resolution and automata



The goal of this paper is to understand the relation between these three methods, using Alternating pushdown systems as a unifying example. The method presented in this paper also extends that presented in [5]. The saturation algorithm of [4] uses only a particular case of Ordered resolution with selection, where only one negative literal can be selected in each clause. In this paper, we generalize the method of [4] to full Ordered resolution with selection. Then, we apply this general method to a formalization of Alternating pushdown systems.

We show, in Section 2, how to express Alternating pushdown systems in Polarized sequent calculus modulo theory. The saturation algorithm is defined in Section 3. We finally apply this saturation algorithm to prove the decidability of inverse reachability in Alternating pushdown systems in Section 4. Due to a lack of space, all proof are omitted. The reader is referred to the long version available on the author's webpages for the details.

2 Alternating pushdown systems in Polarized sequent calculus modulo theory

2.1 Alternating pushdown systems

Definition 1 (State, word, configuration). *Consider a language \mathcal{L} in monadic predicate logic, containing a finite number of predicate symbols, called states, a finite number of function symbols, called stack symbols, and a constant ε , called the empty word. A closed term in \mathcal{L} , also called a word, has the form $\gamma_1(\gamma_2(\dots(\gamma_n(\varepsilon)))$ where $\gamma_1, \dots, \gamma_n$ are stack symbols. A closed atomic proposition, also called a configuration, has the form $P(w)$ where P is a state and w a word.*

Definition 2 (Alternating pushdown systems). *An Alternating pushdown system consists of a finite set of rules of the form*

pop

$$Q(\gamma(x)) \leftrightarrow \{P_1(x), \dots, P_n(x)\}$$

empty

$$Q(\varepsilon) \leftrightarrow \emptyset$$

push

$$Q(x) \leftrightarrow \{P_1(\gamma(x)), P_2(x), \dots, P_n(x)\}$$

and neutral

$$Q(x) \leftrightarrow \{P_1(x), \dots, P_n(x)\}$$

It defines a transition system between finite sets of closed atomic propositions: for all finite set S of closed atomic propositions and all word w

- a pop rule $Q(\gamma(x)) \leftrightarrow \{P_1(x), \dots, P_n(x)\}$ defines a transition

$$\{Q(\gamma(w))\} \cup S \leftrightarrow \{P_1(w), \dots, P_n(w)\} \cup S$$

- an empty rule $Q(\varepsilon) \leftrightarrow \emptyset$ defines a transition

$$\{Q(\varepsilon)\} \cup S \leftrightarrow S$$

- a push rule $Q(x) \leftrightarrow \{P_1(\gamma(x)), P_2(x), \dots, P_n(x)\}$ defines a transition

$$\{Q(w)\} \cup S \leftrightarrow \{P_1(\gamma(w)), P_2(w), \dots, P_n(w)\} \cup S$$

- a neutral rule $Q(x) \leftrightarrow \{P_1(x), \dots, P_n(x)\}$ defines a transition

$$\{Q(w)\} \cup S \leftrightarrow \{P_1(w), \dots, P_n(w)\} \cup S$$

A closed atomic proposition $P(w)$ is *inverse reachable*, written $P(w) \in Pre^*(\emptyset)$, if $\{P(w)\} \leftrightarrow^* \emptyset$, where, as usual, the relation \leftrightarrow^* is the reflexive-transitive closure of the relation \leftrightarrow .

The decidability of such a system is not obvious as, for instance, the push rule

$$P(x) \leftrightarrow \{P(a(x))\}$$

transforms the set $\{P(a(\varepsilon))\}$ successively into $\{P(a(a(\varepsilon)))\}$, $\{P(a(a(a(\varepsilon))))\}$, $\{P(a(a(a(a(\varepsilon))))\}$, and so on.

2.2 Polarized sequent calculus modulo theory

Polarized sequent calculus modulo theory [8, 9] is an extension of sequent calculus, where axioms are replaced by rewrite rules, for instance the axiom $P \Rightarrow (Q \wedge R)$ can be replaced by the rewrite rule $P \longrightarrow_- (Q \wedge R)$. Rules are classified into positive and negative, according to the occurrences of the proposition to which they can apply. Then, the deduction rules are modified to allow the reduction of a proposition at any time.

Definition 3 (Polarized rewrite system). A rewrite rule is a pair $P \longrightarrow A$ where P is an atomic proposition and A an arbitrary proposition. A rewrite system is a set of rewrite rules. A polarized rewrite system $\langle \mathcal{R}_-, \mathcal{R}_+ \rangle$ is a pair of rewrite systems. The rules of \mathcal{R}_- are called negative and those of \mathcal{R}_+ positive.

Definition 4 (Polarized rewriting). Let $\mathcal{R} = \langle \mathcal{R}_-, \mathcal{R}_+ \rangle$ be a polarized rewrite system. The one step negative and positive rewriting relations, \longrightarrow_- and \longrightarrow_+ , are defined as follows.

- If $P \longrightarrow A$ is a rule of \mathcal{R}_s and σ is a substitution then $\sigma P \longrightarrow_s \sigma A$, where s is either $-$ or $+$.
- If $A \longrightarrow_{\bar{s}} A'$ then $\neg A \longrightarrow_s \neg A'$, where $\bar{\cdot}$ swaps $-$ and $+$.
- If $(A \longrightarrow_s A' \text{ and } B = B')$ or $(A = A' \text{ and } B \longrightarrow_s B')$, then $A \wedge B \longrightarrow_s A' \wedge B'$ and $A \vee B \longrightarrow_s A' \vee B'$.
- If $(A \longrightarrow_{\bar{s}} A' \text{ and } B = B')$ or $(A = A' \text{ and } B \longrightarrow_s B')$, then $A \Rightarrow B \longrightarrow_s A' \Rightarrow B'$.
- If $A \longrightarrow_s A'$ then $\forall x A \longrightarrow_s \forall x A'$ and $\exists x A \longrightarrow_s \exists x A'$.

As usual, the relation \longrightarrow_s^* is the reflexive-transitive closure of the relation \longrightarrow_s .

Definition 5 (Polarized sequent calculus modulo theory). Let $\langle \mathcal{R}_-, \mathcal{R}_+ \rangle$ be a polarized rewrite system. Languages, terms, and propositions of Polarized sequent calculus modulo $\langle \mathcal{R}_-, \mathcal{R}_+ \rangle$, are like those of Predicate logic. The deduction rules (Figure 1) are those of the usual sequent calculus parametrized by the polarized rewrite system $\langle \mathcal{R}_-, \mathcal{R}_+ \rangle$.

For instance, the usual rule of sequent calculus

$$\frac{\Gamma \vdash C, \Delta \quad \Gamma \vdash D, \Delta}{\Gamma \vdash C \wedge D, \Delta} \wedge\text{-right}$$

is transformed into

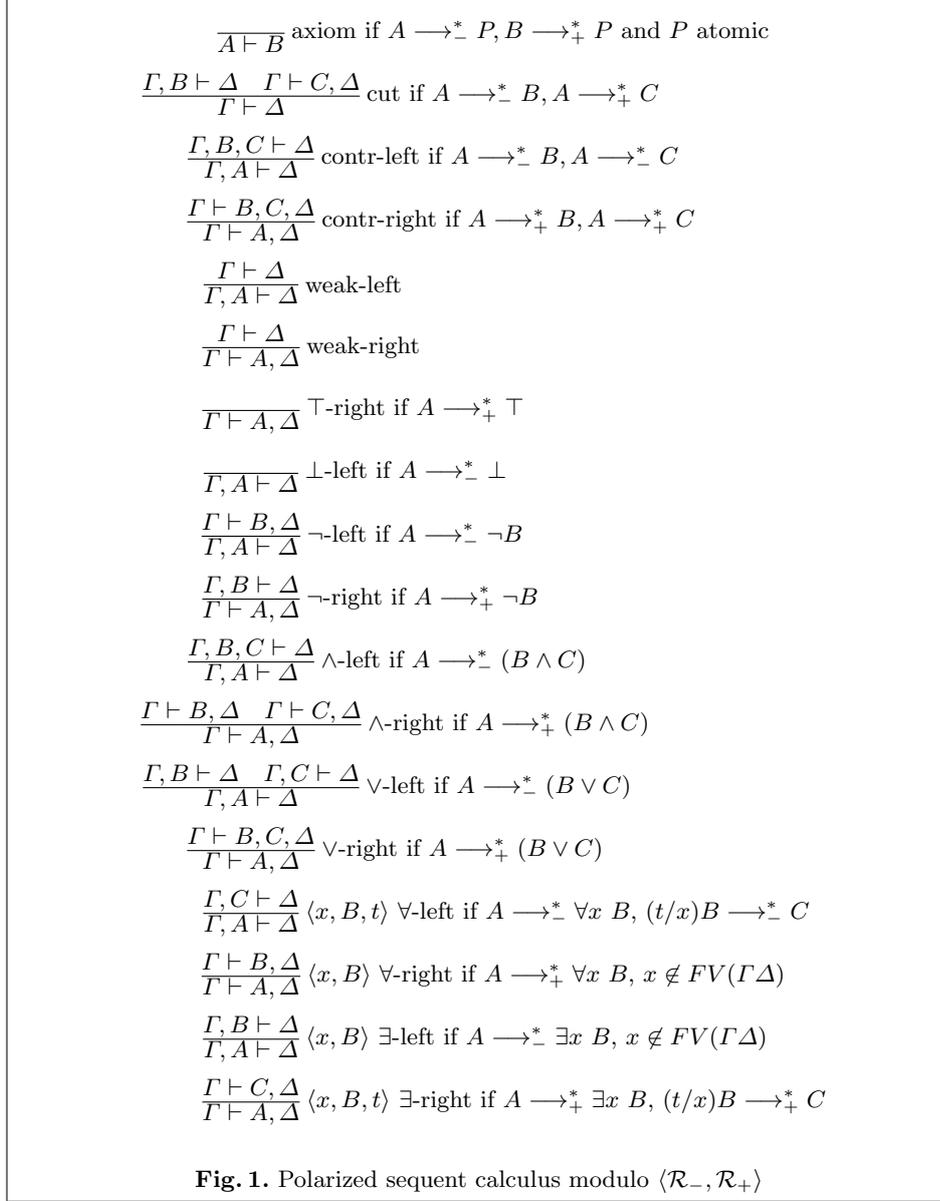
$$\frac{\Gamma \vdash C, \Delta \quad \Gamma \vdash D, \Delta}{\Gamma \vdash A, \Delta} \wedge\text{-right} \quad \text{if } A \longrightarrow_+^* (C \wedge D)$$

Provability is preserved (also when considering only cut-free proofs) if the right-hand sides of rewrite rules is replaced by equivalent propositions. This can be used to simplify rewrite rules, using for instance De Morgan's rules.

Definition 6 (Right-equivalent rewrite rules). Two rewrite rules $P \longrightarrow A$ and $P \longrightarrow B$ are right-equivalent if the proposition $\forall x_1 \dots \forall x_n (A \Leftrightarrow B)$ is a provable in Predicate logic, where x_1, \dots, x_n are the free variables of A and B .

Two rewrite systems are right-equivalent if for each rule in one system, there is a right-equivalent rule in the other. This extends to polarized rewrite systems in the obvious way.

Lemma 1. Given two polarized rewrite systems $\langle \mathcal{R}_-, \mathcal{R}_+^1 \rangle$ and $\langle \mathcal{R}_-, \mathcal{R}_+^2 \rangle$ that are right-equivalent, a sequent $\Gamma \vdash \Delta$ can be proved (resp. proved without cut) modulo $\langle \mathcal{R}_-, \mathcal{R}_+^1 \rangle$ if and only if it can be proved (resp. proved without cut) modulo $\langle \mathcal{R}_-, \mathcal{R}_+^2 \rangle$.



Definition 7 (Simplification). *The simplification $A \downarrow$ of a proposition A is obtained by pushing the negations inside A .*

$$\begin{array}{lll}
(\neg \top) \downarrow = \perp & (\neg(A \wedge B)) \downarrow = (\neg A) \downarrow \vee (\neg B) \downarrow & (A \wedge B) \downarrow = A \downarrow \wedge B \downarrow \\
\top \downarrow = \top & (\neg(A \vee B)) \downarrow = (\neg A) \downarrow \wedge (\neg B) \downarrow & (A \vee B) \downarrow = A \downarrow \vee B \downarrow \\
(\neg \perp) \downarrow = \top & (\neg \forall x A) \downarrow = \exists x ((\neg A) \downarrow) & (\forall x A) \downarrow = \forall x (A \downarrow) \\
\perp \downarrow = \perp & (\neg \exists x A) \downarrow = \forall x ((\neg A) \downarrow) & (\exists x A) \downarrow = \exists x (A \downarrow) \\
(\neg \neg A) \downarrow = A \downarrow & &
\end{array}$$

This definition extends to rewrite rules $(P \longrightarrow A) \downarrow = P \longrightarrow (A \downarrow)$ and to rewrite systems and polarized rewrite systems in the obvious way.

Lemma 2. *Given a polarized rewrite systems $\langle \mathcal{R}_-, \mathcal{R}_+ \rangle$ and its simplification $\langle \mathcal{R}_-, \mathcal{R}_+ \rangle \downarrow$, a sequent $\Gamma \vdash \Delta$ can be proved (resp. proved without cut) modulo $\langle \mathcal{R}_-, \mathcal{R}_+ \rangle$ if and only if it can be proved (resp. proved without cut) modulo $\langle \mathcal{R}_-, \mathcal{R}_+ \rangle \downarrow$.*

2.3 Alternating pushdown systems in Polarized sequent calculus modulo theory

To express Alternating pushdown systems in Polarized sequent calculus modulo theory, we must replace each transition rule of an Alternating pushdown system by a rewrite rule. Let us start with an example.

Example 1. Consider the Alternating pushdown system formed with the rules

$$\begin{array}{ll} P(\varepsilon) \hookrightarrow \emptyset & Q(b(x)) \hookrightarrow \{P(x)\} \\ R(a(x)) \hookrightarrow \{Q(x)\} & S(x) \hookrightarrow \{R(a(x))\} \end{array}$$

The pop rules $R(a(x)) \hookrightarrow \{Q(x)\}$ and $Q(b(x)) \hookrightarrow \{P(x)\}$ and the empty rule $P(\varepsilon) \hookrightarrow \emptyset$ that permit to build the derivation

$$\{R(a(b(\varepsilon)))\} \hookrightarrow \{Q(b(\varepsilon))\} \hookrightarrow \{P(\varepsilon)\} \hookrightarrow \emptyset$$

are transformed into positives rule $R(a(x)) \longrightarrow_+ Q(x)$, $Q(b(x)) \longrightarrow_+ P(x)$, and $P(\varepsilon) \longrightarrow_+ \top$. This way, the proposition $R(a(b(\varepsilon)))$ rewrites positively to \top , and it has the proof in Polarized sequent calculus modulo theory

$$\frac{}{\vdash R(a(b(\varepsilon)))} \top\text{-right}$$

A push rule such as $S(x) \hookrightarrow \{R(a(x))\}$ could be transformed into the positive rule $S(x) \longrightarrow_+ R(a(x))$. This way, the proposition $S(b(\varepsilon))$ would rewrite positively to \top as well, and would have a trivial proof, built with the \top -right rule. However, we prefer to transform it into the negative rule $R(a(x)) \longrightarrow_- S(x)$, in order to let the proposition $S(b(\varepsilon))$ have the proof

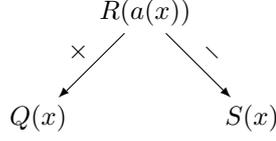
$$\frac{\frac{}{\vdash R(a(b(\varepsilon)))} \top\text{-right} \quad \frac{}{R(a(b(\varepsilon))) \vdash S(b(\varepsilon))} \text{axiom}}{\vdash S(b(\varepsilon))} \text{cut}$$

but no cut-free proof, reflecting the fact that the derivation

$$\{S(b(\varepsilon))\} \hookrightarrow \{R(a(b(\varepsilon)))\} \hookrightarrow \{Q(b(\varepsilon))\} \hookrightarrow \{P(\varepsilon)\} \hookrightarrow \emptyset$$

contains a ‘‘cut’’: a subsequence formed with a push rule followed by a pop rule. With the rewrite rules we have so far, this cut cannot be eliminated because the

critical pair



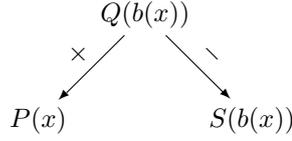
does not close. But, it can be eliminated if we add either the rule $S(x) \rightarrow_+ Q(x)$ or the rule $Q(x) \rightarrow_- S(x)$, both representing the neutral rule $S(x) \leftrightarrow \{Q(x)\}$. Again, we shall take the negative rule, so that the closed atomic proposition $S(b(\varepsilon))$ has a proof

$$\frac{\frac{}{\vdash Q(b(\varepsilon))} \top\text{-right} \quad \frac{}{Q(b(\varepsilon)) \vdash S(b(\varepsilon))} \text{axiom}}{\vdash S(b(\varepsilon))} \text{cut}$$

but no cut-free proof, reflecting the fact that the derivation

$$\{S(b(\varepsilon))\} \leftrightarrow \{Q(b(\varepsilon))\} \leftrightarrow \{P(\varepsilon)\} \leftrightarrow \emptyset$$

still contains a “cut”: a subsequence formed with a neutral rule followed by a pop rule. With the rewrite rules we have so far, this cut cannot be eliminated because the critical pair



does not close. But, it can be eliminated if we add the rule $S(b(x)) \rightarrow_+ P(x)$ which represents the pop rule $S(b(x)) \leftrightarrow \{P(x)\}$.

Once these two rules have been added, the proposition $S(b(\varepsilon))$ rewrites positively to \top , $S(b(\varepsilon)) \rightarrow_+ P(\varepsilon) \rightarrow_+ \top$, and hence it has the cut-free proof

$$\frac{}{\vdash S(b(\varepsilon))} \top\text{-right}$$

More generally, we shall see that with these two extra rules, every provable proposition has a cut-free proof and that the existence of a cut-free proof is decidable.

This example leads to the following definition.

Definition 8 (Expression of Alternating pushdown systems). *Let \mathcal{P} be an Alternating pushdown system, we define the rewrite system $\mathcal{R}(\mathcal{P})$ as follows:*

- a pop rule $Q(\gamma(x)) \leftrightarrow \{P_1(x), \dots, P_n(x)\}$ is expressed as the rewrite rule

$$Q(\gamma(x)) \rightarrow_+ P_1(x) \wedge \dots \wedge P_n(x)$$

– an empty rule $Q(\varepsilon) \leftrightarrow \emptyset$ as the rewrite rule

$$Q(\varepsilon) \longrightarrow_+ \top$$

– a push rule $Q(x) \leftrightarrow \{P_1(\gamma(x)), P_2(x), \dots, P_n(x)\}$ as the rewrite rule

$$P_1(\gamma(x)) \longrightarrow_- \neg P_2(x) \vee \dots \vee \neg P_n(x) \vee Q(x)$$

– a neutral rule $Q(x) \leftrightarrow \{P_1(x), \dots, P_n(x)\}$ with $n \geq 1$ as the n rewrite rules

$$P_i(x) \longrightarrow_- \neg P_1(x) \vee \dots \vee \neg P_{i-1}(x) \vee \neg P_{i+1}(x) \dots \vee \neg P_n(x) \vee Q(x)$$

– and a neutral rule $Q(x) \leftrightarrow \emptyset$ as the rewrite rule

$$Q(x) \longrightarrow_+ \top$$

Lemma 3. *Let \mathcal{P} be an Alternating pushdown system and $\mathcal{R} = \mathcal{R}(\mathcal{P})$ be its expression as a rewrite system. If $S \leftrightarrow^* \emptyset$ in \mathcal{P} , then for all A in S , the sequent $\vdash A$ has a proof in Polarized sequent calculus modulo \mathcal{R} .*

3 Ordered resolution with selection and cut-elimination

We now want to define a saturation algorithm, that transforms a polarized rewrite system into an equivalent *saturated* polarized rewrite system and prove that if a polarized rewrite system \mathcal{R} is saturated, then the Polarized sequent calculus modulo \mathcal{R} has the cut-elimination property. Instead of saturating a polarized rewrite system by closing the critical pairs one by one, we shall use a more powerful method [4]: transform a polarized rewrite system into a set of clauses, saturate this set of clauses with Ordered resolution with selection and transform back the saturated set of clauses into a polarized rewrite system. Note that in general, saturating using Ordered resolution with selection may not terminate, so that not all rewrite systems can be completed that way. However, we will see cases where saturation is guaranteed to terminate.

3.1 Ordered resolution with selection

Ordered resolution with selection is a refinement of Resolution parametrized by a Noetherian ordering and a selection function. We recall its definition.

Definition 9 (Literal, Clause). *A proposition is a literal if it is either atomic or the negation of an atomic proposition. A clause is a disjunction of literals.*

If A is a proposition containing the free variables x_1, \dots, x_n , we write $\bar{\forall}A$ for the proposition $\forall x_1 \dots \forall x_n A$.

Definition 10 (Clause form). *To each proposition A , we associate a set of clauses $\mathcal{C}(A)$ such that A is equivalent to $\bigwedge_{C \in \mathcal{C}(A)} \bar{\forall}C$. The actual definition of \mathcal{C} is not relevant here, see for instance [16].*

Definition 11 (Maximal atomic proposition). Given an ordering \succ on atomic propositions, an atomic proposition A is said to be maximal in a clause C if for all atomic propositions B in C , $B \not\succeq A$.

Definition 12 (Selection function). A selection function is a function \mathcal{S} that associates to each clause C a set $\mathcal{S}(C)$ of negative literals of C .

We often indicate the selected literals just by underlining them; for instance, in the clause $\neg P \vee \neg Q \vee \neg R \vee S$, if \mathcal{S} selects $\neg P$ and $\neg R$, we write the clause $\underline{\neg P} \vee \neg Q \vee \underline{\neg R} \vee S$.

Definition 13 (Ordered resolution with selection). Ordered resolution with selection [2, Section 4.3] (ORS_S^\succ) is a refinement of Resolution parametrized by

- an ordering \succ on atomic propositions that is well-founded, stable by substitution, and total on ground atomic propositions,
- and a selection function \mathcal{S} .

It consists in restricting the literals on which the Resolution rule can be applied: if $\mathcal{S}(C)$ is not empty, then only the literals in $\mathcal{S}(C)$ can be used; otherwise, only the maximal literals with respect to \succ can be used.

The inference rule of ORS_S^\succ is the following:

$$\frac{\neg A_1 \vee \dots \vee \neg A_k \vee D \quad A_1^1 \vee \dots \vee A_1^{n_1} \vee C_1 \quad \dots \quad A_k^1 \vee \dots \vee A_k^{n_k} \vee C_k}{\sigma(D \vee C_1 \vee \dots \vee C_k)} \text{Resolution}$$

- σ is the most general unifier to the simultaneous unification problems $A_i^1 = \dots = A_i^{n_i} = A_i$ for $1 \leq i \leq k$;
- - either $\mathcal{S}(\neg A_1 \vee \dots \vee \neg A_k \vee D) = \{\neg A_1, \dots, \neg A_k\}$, with $k \geq 1$;
 - or $\mathcal{S}(\neg A_1 \vee \dots \vee \neg A_k \vee D) = \emptyset$, $k = 1$ and $\sigma(\neg A_1)$ is maximal in $\sigma(\neg A_1 \vee D)$;
- $\mathcal{S}(A_i^1 \vee \dots \vee A_i^{n_i} \vee C_i) = \emptyset$ for all $1 \leq i \leq k$;
- and each $\sigma(A_i^j)$ is maximal in $\sigma(A_i^1 \vee \dots \vee A_i^{n_i} \vee C_i)$.

Definition 14 (Saturated). A set of clauses \mathcal{C} is saturated if no clause can be added by applying the Resolution rule.

Remark 1. The results presented here hold also if one considers saturation up to redundancies, more precisely up to compositeness [1], where inferred clauses can be discarded from the saturation if they follow from smaller clauses.

Besides, if \mathcal{C} and \mathcal{D} are two sets of clauses and \mathcal{C} is saturated, then in a derivation of \perp from \mathcal{C}, \mathcal{D} , we can forbid applying a rule between clauses from \mathcal{C} only, because this would yield a clause that is already in \mathcal{C} .

3.2 Building clauses from polarized rewrite systems

Definition 15 (Transforming polarized rewrite systems into sets of clauses). Let \mathcal{R} be a polarized rewrite system. Let Γ be the set of propositions containing, for each negative rewrite rule $P \rightarrow_- A$ in \mathcal{R} , the proposition $\bar{\forall}(P \Rightarrow A)$ and, for each positive rewrite rule $P \rightarrow_+ A$ in \mathcal{R} , the proposition $\bar{\forall}(A \Rightarrow P)$. The set of clauses $\Phi(\mathcal{R})$ associated to the polarized rewrite system \mathcal{R} is the set $\bigcup_{B \in \Gamma} \mathcal{C}(B)$.

Lemma 4. *Let \mathcal{R} be a polarized rewrite system and Γ be as in Definition 15. Let $\{C_1, \dots, C_n\} = \Phi(\mathcal{R})$ be the set of clauses associated to \mathcal{R} . Let D be a proposition. Then, the sequent $\vdash D$ is provable in Polarized sequent calculus modulo \mathcal{R} if and only if the sequent $\Gamma \vdash D$ is provable in Predicate logic if and only if the sequent $\forall C_1, \dots, \forall C_n \vdash D$ is provable in Predicate logic.*

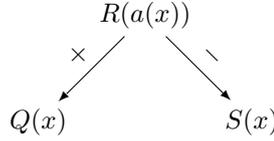
Example 2. Continuing Example 1, the polarized rewrite system

$$\begin{array}{ll} P(\varepsilon) \longrightarrow_+ \top & Q(b(x)) \longrightarrow_+ P(x) \\ R(a(x)) \longrightarrow_+ Q(x) & R(a(x)) \longrightarrow_- S(x) \end{array}$$

is transformed into the set of clauses

$$\begin{array}{ll} P(\varepsilon) & \neg P(x) \vee Q(b(x)) \\ \neg Q(x) \vee R(a(x)) & \neg R(a(x)) \vee S(x) \end{array}$$

The rewrite rule saturation step of Example 1, that generates the rewrite rule $Q(x) \longrightarrow_- S(x)$ closing the critical pair



is the application of the Ordered resolution rule —with an ordering \succ such that $R(a(x)) \succ Q(x)$, $R(a(x)) \succ S(x)$ — to two clauses $\neg Q(x) \vee R(a(x))$ and $\neg R(a(x)) \vee S(x)$, that yields the clause $\neg Q(x) \vee S(x)$. Then, in this clause, we need to consider the negative literal $\neg Q(x)$ as "maximal", so that we can apply the Resolution rule to the clauses $\neg Q(x) \vee S(x)$ and $\neg P(x) \vee Q(b(x))$ to generate the clause $S(b(x)) \vee \neg P(x)$ corresponding to the rule $S(b(x)) \longrightarrow_+ P(x)$ of Example 1. To do so, we must use the selection function to select the negative literal $\neg Q(x)$ in the clause $\neg Q(x) \vee S(x)$ corresponding to a neutral rule.

3.3 Building polarized rewrite systems from clauses using Ordered resolution with selection

In [4], we have presented an algorithm that transforms a set of clauses \mathcal{C} into a polarized rewrite system \mathcal{R} , such that Polarized sequent calculus modulo \mathcal{R} has the cut-elimination property. More precisely, this algorithm first saturates this set of clauses \mathcal{C} , with Ordered resolution with selection, into an equivalent set of clauses \mathcal{C}' . If this saturation terminates, the algorithm generates the polarized rewrite system \mathcal{R} , from the saturated set \mathcal{C}' . However [4] only addressed a restricted form of Ordered resolution with selection, where in each clause, at most one literal is selected. So, in this section, we start by generalizing the main theorem of [4] to full Ordered resolution with selection.

Definition 16 (Polarized rewrite system associated to a set of clauses). *Given an ordering \succ on atomic propositions and a selection function \mathcal{S} , we define a function $\Psi_{\mathcal{S}}^{\succ}$ that associates, to each set of clauses \mathcal{C} , the following polarized rewrite system $\Psi_{\mathcal{S}}^{\succ}(\mathcal{C})$: for each clause C of \mathcal{C} ,*

– if $\mathcal{S}(C) \neq \emptyset$, then for all $\neg Q \in \mathcal{S}(C)$, we add a rule

$$Q \longrightarrow_{-} \forall y_1 \dots \forall y_n (C \setminus \neg Q)$$

– if $\mathcal{S}(C) = \emptyset$, then

- for all maximal negative literals $\neg Q$ in C , we add a rule

$$Q \longrightarrow_{-} \forall y_1 \dots \forall y_n (C \setminus \neg Q)$$

- for all maximal positive literals Q in C , we add a rule

$$Q \longrightarrow_{+} (\neg \forall y_1 \dots \forall y_n (C \setminus Q)) \downarrow$$

where $C \setminus L$ is the clause C without the literal L and y_1, \dots, y_n are the variables that are free in C but not in Q .

Lemma 5. *The transformation $\Psi_S^>$ is a right inverse (in terms of logical equivalence) of the function Φ of Definition 15: for all set S of clauses, $\Phi(\Psi_S^>(S))$ is logically equivalent to S .*

Note that it is not a left inverse (because the left-hand side of the original rules are not necessarily selected or maximal in the clauses).

Example 3. Consider the polarized rewrite system containing the rules $P \longrightarrow_{-} Q \wedge R$ and $P \longrightarrow_{+} \top$. The proposition Q has no cut-free proof, but it has the proof

$$\frac{\frac{\frac{}{\vdash P} \top\text{-right}}{\vdash P} \quad \frac{\frac{Q, R \vdash Q}{P \vdash Q} \text{axiom}}{\vdash Q} \wedge\text{-left}}{\vdash Q} \text{cut}}$$

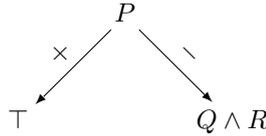
The set of clauses associated to this rewrite system is $\{\neg P \vee Q, \neg P \vee R, P\}$. If we consider an order such that $R \succ Q \succ P$, and a selection function selecting $\neg P$ in both clauses, then Resolution generates the clauses Q and R and the produced rewrite system is

$$\begin{array}{ccc} P \longrightarrow_{-} Q & P \longrightarrow_{+} \top & R \longrightarrow_{+} \top \\ P \longrightarrow_{-} R & Q \longrightarrow_{+} \top & \end{array}$$

In contrast, if, with the same order, selection does not select any negative literal, Resolution does not generate any clause, and the produced rewrite system is

$$Q \longrightarrow_{+} P \quad R \longrightarrow_{+} P \quad P \longrightarrow_{+} \top$$

The proposition Q has a cut-free proof in both cases, but the two rewrite systems are quite different. This example also shows that it is not completely obvious to saturate directly the rewrite system and add such rules as a way to close the critical pair



Example 4. Continuing on Examples 1 and 2, using the order and selection function defined there, the clause $P(\varepsilon)$, $\neg P(x) \vee Q(b(x))$, $\neg Q(x) \vee R(a(x))$, and $\neg R(a(x)) \vee S(x)$ of the initial system translate respectively to the rewrite rules

$$\begin{array}{ll} P(\varepsilon) \longrightarrow_+ \top & Q(b(x)) \longrightarrow_+ P(x) \\ R(a(x)) \longrightarrow_+ Q(x) & R(a(x)) \longrightarrow_- S(x) \end{array}$$

and the clause $\neg Q(x) \vee S(x)$ and $\neg P(x) \vee S(b(x))$ added during saturation translate respectively to the rewrite rules

$$Q(x) \longrightarrow_- S(x) \quad S(b(x)) \longrightarrow_+ P(x)$$

To generalize the main theorem of [4] linking saturation with cut elimination, we introduce another variant of Resolution, with a binary resolution rule and an explicit factoring rule.

Definition 17 (Binary resolution and factoring).

$$\frac{P \vee C \quad \neg Q \vee D}{\sigma(C \vee D)} \text{ Binary Resolution} \quad \frac{P \vee Q \vee C}{\sigma(P \vee C)} \text{ Factoring}$$

where σ is the most general unifier of P and Q .

Theorem 1 (Cut-elimination). *If $\mathcal{C} = \{C_1, \dots, C_n\}$ is a saturated set of clauses and A is an arbitrary proposition, then the following are equivalent:*

1. *the sequent $\vdash A$ is provable in the Polarized sequent calculus modulo $\Psi_{\mathcal{S}}^{\succ}(\mathcal{C})$;*
2. *the sequent $\forall C_1, \dots, \forall C_n \vdash A$ is provable in Predicate logic;*
3. *the empty clause can be derived in $ORS_{\mathcal{S}}^{\succ}$ from $\mathcal{C}, \mathcal{C}(\neg A)$, with, at each step, at least one of the premises not in \mathcal{C} ;*
4. *the empty clause can be derived in the Binary Resolution and Factoring system from $\mathcal{C}, \mathcal{C}(\neg A)$, with, at each step, at least one of the premises not in \mathcal{C} (so Factoring is applied only to clauses outside \mathcal{C});*
5. *the sequent $\mathcal{C}(\neg A) \vdash$ has a cut-free proof in the Polarized sequent calculus modulo $\Psi_{\mathcal{S}}^{\succ}(\mathcal{C})$;*
6. *the sequent $\vdash A$ has a cut-free proof in the Polarized sequent calculus modulo $\Psi_{\mathcal{S}}^{\succ}(\mathcal{C})$.*

Corollary 1. *If \mathcal{C} is saturated, then the cut rule is admissible in Polarized sequent calculus modulo $\Psi_{\mathcal{S}}^{\succ}(\mathcal{C})$.*

4 An application to the decidability of Alternating pushdown systems

The goal of this section is to prove the decidability of reachability in pushdown systems (Theorem 2) by using to the result of Section 3. We have shown in Section 2 how to express an Alternating pushdown system \mathcal{P} as a polarized rewrite system $\mathcal{R}(\mathcal{P})$. When we express this rewrite system as a set of clauses $\Phi(\mathcal{R}(\mathcal{P}))$, we obtain *Pushdown clauses*.

Definition 18 (Pushdown clause). Pushdown clauses are clauses of the following form:

$$\begin{array}{lll}
\neg Q_1(x) \vee \dots \vee \neg Q_n(x) \vee P(\gamma(x)) & \text{pop} & \text{where } n \text{ may be zero} \\
P(\varepsilon) & \text{empty} & \\
\neg P_1(\gamma(x)) \vee \neg P_2(x) \vee \dots \vee \neg P_n(x) \vee Q(x) & \text{push} & \\
\neg Q_1(x) \vee \dots \vee \neg Q_n(x) \vee P(x) & \text{neutral} & \text{where } n \geq 1 \\
P(x) & \text{arbitrary} &
\end{array}$$

Note that we distinguish the clauses coming from neutral rules with at least one premise (neutral clauses) and those coming from neutral rules with no premise (arbitrary clauses).

Definition 19 (Coherent). Let \mathcal{P} be an Alternating pushdown system. A clause C of the form $\neg P_1 \vee \dots \vee \neg P_n \vee Q$ is said to be coherent with the Alternating pushdown system \mathcal{P} if for all substitutions σ , such that σC is closed, if $\{\sigma P_1\} \hookrightarrow^* \emptyset$, ..., $\{\sigma P_n\} \hookrightarrow^* \emptyset$ in \mathcal{P} , then $\{\sigma Q\} \hookrightarrow^* \emptyset$ in \mathcal{P} .

Lemma 6. Let \mathcal{P} be an Alternating pushdown system, all the clauses of $\Phi(\mathcal{R}(\mathcal{P}))$ are coherent with \mathcal{P} .

Definition 20 (Ordering). Let \prec be the Knuth-Bendix Ordering [15] with an arbitrary total precedence and the same weight for each symbol.

Note that $P(\gamma(x)) \succ Q(x)$ regardless the predicate symbols P and Q and that \prec is total on closed atomic propositions. Any order satisfying these two conditions could be used instead.

Definition 21 (Selection). Consider a selection function S that selects all the negative literals in neutral clauses, and nothing in pop, empty, push, and arbitrary clauses.

A clause C such that $S(C) \neq \emptyset$ is a neutral clause. A clause C such that $S(C) = \emptyset$ and C has a negative maximal literal is an push clause. A clause C such that $S(C) = \emptyset$ and C has a positive maximal literal is either an pop clause, an empty clause, or an arbitrary clause.

Thus, when applying the Resolution rule to Pushdown clauses (Definition 18) the clause with the negative literals can either be an push clause or a neutral clause, and those with the positive literals can be pop, empty, or arbitrary clauses.

If the clause with negative literals is an push clause, the clause with positive literals can be a pop clause or an arbitrary clause, but not an empty clause, because ε and $\gamma(x)$ do not unify. Thus, the resolution rules specializes into the two rules

$$\begin{array}{l}
\frac{\neg Q_1(\gamma(x)) \vee \neg Q_2(x) \vee \dots \vee \neg Q_m(x) \vee R(x) \quad \neg P_1(x) \vee \dots \vee \neg P_n(x) \vee Q_1(\gamma(x))}{\neg Q_2(x) \vee \dots \vee \neg Q_m(x) \vee \neg P_1(x) \vee \dots \vee \neg P_n(x) \vee R(x)} \text{push-pop} \\
\frac{\neg Q_1(\gamma(x)) \vee \neg Q_2(x) \vee \dots \vee \neg Q_m(x) \vee R(x) \quad Q_1(x)}{\neg Q_2(x) \vee \dots \vee \neg Q_m(x) \vee R(x)} \text{push-arbitrary}
\end{array}$$

If the clause with negative literals is a neutral clause, then the clauses with positive literals are a mix of pop, empty, and arbitrary clauses. There cannot be both pop and empty clauses, because ε and $\gamma(x)$ do not unify, so they are either a mix of at least one pop clause and some arbitrary clauses, a mix of at least one empty clauses and some arbitrary clauses, or arbitrary clauses only. Thus, the resolution rules specializes into the three rules

$$\frac{\frac{\frac{\frac{\neg Q_1(x) \vee \dots \vee \neg Q_k(x) \vee \neg Q_{k+1}(x) \vee \dots \vee \neg Q_m(x) \vee R(x)}{\neg P_1^1(x) \vee \dots \vee \neg P_{n_1}^1(x) \vee Q_1(\gamma(x))} \quad \dots \quad \frac{\neg P_1^k(x) \vee \dots \vee \neg P_{n_k}^k(x) \vee Q_k(\gamma(x))}{Q_{k+1}(x)} \quad \dots \quad \frac{Q_m(x)}{Q_m(x)}}{\neg P_1^1(x) \vee \dots \vee \neg P_{n_1}^1(x) \vee \dots \vee \neg P_1^k(x) \vee \dots \vee \neg P_{n_k}^k(x) \vee R(\gamma(x))} \text{neutral-pop}}{\frac{\frac{\neg Q_1(x) \vee \dots \vee \neg Q_k(x) \vee \neg Q_{k+1}(x) \vee \dots \vee \neg Q_m(x) \vee R(x)}{R(\varepsilon)} \quad Q_1(\varepsilon) \quad \dots \quad Q_k(\varepsilon) \quad Q_{k+1}(x) \quad \dots \quad Q_m(x)}{R(\varepsilon)} \text{neutral-empty}}{\frac{\frac{\neg Q_1(x) \vee \dots \vee \neg Q_m(x) \vee R(x)}{R(x)} \quad Q_1(x) \quad \dots \quad Q_m(x)}{R(x)} \text{neutral-arbitrary}} \text{neutral-empty}$$

These rules are similar to the saturation rules of [10], with one exception: if we have the clauses $P(x)$ and $\neg P(\gamma(x)) \vee Q(x)$, [10] produces first the *pop* clause $P(\gamma(x))$ then $Q(x)$, while Ordered resolution with selection produces $Q(x)$ directly, with the push-arbitrary rule.

Lemma 7. *The Resolution rule applied to Pushdown clauses always produces Pushdown clauses, and terminates, on an saturated set, after a finite number of steps.*

Lemma 8. *Consider an Alternating pushdown system \mathcal{P} . The set of clauses that are coherent with \mathcal{P} is closed by the Resolution rule.*

Proposition 1. *Consider an Alternating pushdown system \mathcal{P} , its expression \mathcal{R} ($= \mathcal{R}(\mathcal{P})$) as a polarized rewrite system, the translation \mathcal{C} ($= \Phi(\mathcal{R})$) of this polarized rewrite system as a set of clauses, the saturation \mathcal{C}' of \mathcal{C} , the polarized rewrite system \mathcal{R}' ($= \Psi_S^>(\mathcal{C}')$) associated to \mathcal{C}' , and the polarized rewrite system \mathcal{R}'' composed of the positive rules of \mathcal{R}' .*

Let $R(w)$ be a closed atomic proposition. Then if $\{R(w)\} \hookrightarrow^ \emptyset$ in \mathcal{P} , the sequent $\vdash R(w)$ has a cut-free proof in the Polarized sequent calculus modulo \mathcal{R}'' .*

We now want to prove the converse of Proposition 1: cut-free proofs in Polarized sequent calculus modulo \mathcal{R}'' can be translated back to derivations in \mathcal{P} .

Lemma 9. *Let \mathcal{R}'' be a polarized rewrite system containing rules of the form*

$$\begin{array}{l} Q(\gamma(x)) \longrightarrow_+ P_1(x) \wedge \dots \wedge P_n(x) \\ Q(\varepsilon) \longrightarrow_+ \top \end{array} \qquad Q(x) \longrightarrow_+ \top$$

and let $R(w)$ be an atomic proposition such that $\vdash R(w)$ has a cut-free proof in Polarized sequent calculus modulo \mathcal{R}'' . Then $R(w) \longrightarrow_+^ T$ where T is a conjunction of propositions \top .*

Proposition 2. Consider $\mathcal{P}, \mathcal{R}, \mathcal{C}, \mathcal{C}', \mathcal{R}'$ and \mathcal{R}'' as in Proposition 1. Let $R(w)$ be a closed atomic proposition. Then if the sequent $\vdash R(w)$ has a cut-free proof in the Polarized sequent calculus modulo \mathcal{R}'' , then $\{R(w)\} \hookrightarrow^* \emptyset$ in \mathcal{P} .

Theorem 2 (Decidability). Consider $\mathcal{P}, \mathcal{R}, \mathcal{C}, \mathcal{C}', \mathcal{R}'$ and \mathcal{R}'' as in Proposition 1. The set of atomic propositions $R(w)$ that have a proof in the Polarized sequent calculus modulo \mathcal{R}'' is decidable.

Example 5. Consider the system with the transition rules

$$\begin{array}{lll} P(a(x)) \hookrightarrow \{P(x), Q(x)\} & P(b(x)) \hookrightarrow \{S(x)\} & Q(b(x)) \hookrightarrow \emptyset \\ Q(a(x)) \hookrightarrow \{S(x)\} & R(x) \hookrightarrow \{P(a(x))\} & S(x) \hookrightarrow \emptyset \end{array}$$

We have $\{R(a(b(\varepsilon)))\} \hookrightarrow^* \emptyset$.

This system is expressed in Polarized sequent calculus modulo theory by the polarized rewrite system

$$\begin{array}{lll} P(a(x)) \longrightarrow_+ P(x) \wedge Q(x) & P(b(x)) \longrightarrow_+ S(x) & Q(b(x)) \longrightarrow_+ \top \\ Q(a(x)) \longrightarrow_+ S(x) & P(a(x)) \longrightarrow_- R(x) & S(x) \longrightarrow_+ \top \end{array}$$

This polarized rewrite system is expressed by the Pushdown clauses \mathcal{C}

$$\begin{array}{lll} \neg P(x) \vee \neg Q(x) \vee P(a(x)) & \neg S(x) \vee P(b(x)) & Q(b(x)) \\ \neg S(x) \vee Q(a(x)) & \neg P(a(x)) \vee R(x) & S(x) \end{array}$$

Applying the Resolution rule to the clauses of \mathcal{C} yields the saturated system \mathcal{C}' that contains the clauses of \mathcal{C} and the following three new clauses

$$\frac{\neg P(x) \vee \neg Q(x) \vee R(x)}{\neg P(x) \vee \neg Q(x) \vee \neg S(x) \vee R(a(x))} \quad \neg S(x) \vee R(b(x))$$

From the set \mathcal{C}' , we derive the following polarized rewrite system

$$\begin{array}{lll} P(a(x)) \longrightarrow_+ P(x) \wedge Q(x) & P(b(x)) \longrightarrow_+ S(x) & Q(b(x)) \longrightarrow_+ \top \\ Q(a(x)) \longrightarrow_+ S(x) & P(a(x)) \longrightarrow_- R(x) & S(x) \longrightarrow_+ \top \\ P(x) \longrightarrow_- \neg Q(x) \vee R(x) & Q(x) \longrightarrow_- \neg P(x) \vee R(x) & \\ R(a(x)) \longrightarrow_+ P(x) \wedge Q(x) \wedge S(x) & R(b(x)) \longrightarrow_+ S(x) & \end{array}$$

In this polarized rewrite system, the proposition $R(a(b(\varepsilon)))$ reduces to the proposition $\top \wedge \top \wedge \top$ and has the cut free proof

$$\frac{\frac{\overline{\vdash \top}}{\vdash \top} \top\text{-right} \quad \frac{\overline{\vdash \top}}{\vdash \top} \top\text{-right} \quad \frac{\overline{\vdash \top}}{\vdash \top} \top\text{-right}}{\vdash R(a(b(\varepsilon)))} \wedge\text{-right}$$

Acknowledgements

This work is supported by the ANR-NSFC project LOCALI (NSFC 61161130530 and ANR 11 IS02 002 01) and the Chinese National Basic Research Program (973) Grant No. 2014CB340302.

References

1. Bachmair, L., Ganzinger, H.: Rewrite-based equational theorem proving with selection and simplification. *Journal of Logic and Computation* 4(3), 1–31 (1994)
2. Bachmair, L., Ganzinger, H.: Resolution theorem proving. In: Robinson and Voronkov [17], pp. 19–99
3. Bouajjani, A., Esparza, J., Maler, O.: Reachability analysis of pushdown automata: Application to model-checking. In: Mazurkiewicz, A.W., Winkowski, J. (eds.) *Concurrency theory*. Lecture Notes in Computer Science, vol. 1243, pp. 135–150 (1997)
4. Burel, G.: Cut admissibility by saturation. In: Dowek, G. (ed.) *RTA-TLCA 2014*. LNCS, vol. 8560, pp. 124–138. Springer (2014)
5. Burel, G., Dowek, G., Jiang, Y.: A completion method to decide reachability in rewrite systems. In: *International Symposium on Frontiers of Combining Systems* (2015)
6. Charatonik, W., Podelski, A.: Set constraints with intersection. *Information and Computation* 179(2), 213–229 (2002)
7. Comon, H., Dauchet, M., Gilleron, R., Jacquemard, F., Lugiez, D., Löding, C., Tison, S., Tommasi, M.: *Tree automata techniques and applications* (2007), <http://tata.gforge.inria.fr/>
8. Dowek, G.: What is a theory? In: *Symposium on Theoretical Aspects of Computer Science*. Lecture Notes in Computer Science, vol. 2285, pp. 50–64 (2002)
9. Dowek, G.: Polarized resolution modulo. In: Calude, C., Sassone, V. (eds.) *IFIP Theoretical Computer Science*. IFIP Advances in Information and Communication Technology, vol. 323, pp. 182–196. Springer (2010)
10. Dowek, G., Jiang, Y.: Cut-elimination and the decidability of reachability in alternating pushdown systems (2014), manuscript
11. Frühwirth, T., Shapiro, E., Vardi, M., Yardeni, E.: Logic programs as types for logic programs. In: *Logic in Computer Science*. pp. 300–309 (1991)
12. Goubault-Larrecq, J.: Deciding \mathcal{H}_1 by resolution. *Information Processing Letters* 95(3), 401–408 (2005)
13. Heintze, N., Jaffar, J.: A decision procedure for a class of set constraints. In: *Logic in Computer Science*. pp. 42–51 (1990)
14. Kleene, S.: *Introduction to Metamathematics*. North-Holland (1952)
15. Knuth, D., Bendix, P.: Simple word problems in universal algebras. In: Leech, J. (ed.) *Computational Problems in Abstract Algebras*. p. 263–297. Pergamon Press (1970)
16. Nonnengart, A., Weidenbach, C.: Computing small clause normal forms. In: Robinson and Voronkov [17], pp. 335–367
17. Robinson, J.A., Voronkov, A. (eds.): *Handbook of Automated Reasoning* (in 2 volumes). Elsevier and MIT Press (2001)