A PSPACE-complete fragment of second-order linear logic

G. Perrier *

LORIA - Université Nancy 2, Campus Scientifique-B.P. 239, 54506 Vandœuvre-les-Nancy Cedex, France

Abstract

Existentially quantified variables are the source of non-decidability for second-order linear logic without exponentials (MALL2). We present a decision procedure for a fragment of MALL2 based on a canonical instantiation of these variables and using inference permutability in proofs. We also establish that this fragment is PSPACE-complete. © 1999 Elsevier Science B.V. All rights reserved.

0. Introduction

The decision problem for second-order linear logic without exponentials (MALL2) has given rise recently to many papers and results. P. Lincoln, A. Scedrov and N. Shankar have shown that the multiplicative fragment of second order intuitionistic linear logic (IML2) is undecidable by encoding second order intuitionistic propositional logic – known to be undecidable – into IML2 [10]. Emms has extended this strategy to prove the undecidability of the second-order Lambek Calculus (L2) [3], which can be viewed as the multiplicative fragment of intuitionistic non-commutative linear logic. The undecidability of MALL2 has been shown by Lafont [6] and the undecidability of the multiplicative fragment of second-order propositional linear logic (MLL2) by Lafont and Scedrov [7], in both cases with an encoding of two-counter machines.

Motivated by linguistic considerations, Emms has shown the decidability of a fragment of L2 [2]. In L2, undecidability comes from universally quantified formulas on the left-hand side of sequents. Emms limits such formulas to five formulas that represent all polymorphic categories according to his view of the Categorial Grammar theory for the syntax of natural languages. For these five formulas, he proposes a canonical method for eliminating their variables and the source of undecidability at the same time.

* E-mail: perrier@loria.fr.
When such a formula is decomposed in a bottom-up proof search, the propositional variable is not instantiated immediately. The decomposition of the formula continues until all occurrences of the variable emerge. At this moment and because of the particular syntax of the decomposed formula, a canonical instantiation of the variable can be found.

We generalise the result of Emms to the broader framework of MALL2. With respect to our problem of decidability, the main difference between L2 and MALL2 comes from the presence of additives in MALL2, the non-commutativity or the commutativity of the logic being irrelevant. If we present MALL2 in the formalism of the one-sided sequent calculus, undecidability comes from the existentially quantified formulas. We exhibit general criteria permitting the method of Emms to be applied to such formulas. A crucial point is to have the possibility of decomposing these formulas at one go in order to make the occurrences of the variable emerge. This implies the possibility of applying the inference rules in a particular order. That is why in Section 1 we begin by a preliminary study of inference permutability in MALL2. This study will guide us in Section 2 in order to define the syntax of the existentially quantified formulas that will belong to the fragment for which a decision procedure will be defined. By implementing this procedure in an economical way, we show that the fragment is PSPACE and, since it includes propositional MALL, which is PSPACE-complete [9], it is also PSPACE-complete.

1. Deduction procedures and inference permutability in MALL2

1.1. Deduction procedures in MALL2

We present the inference system of MALL2 in the framework of the one-sided sequent calculus. Sequents have the form \( \Gamma \vdash \Delta \) where \( \Delta \) is a finite multi-set of MALL2 formulas.

To define MALL2 formulas, we assume a countably infinite set \( \mathcal{V} \) of propositional variables and a countably infinite set \( \mathcal{C} \) of propositional constants. \( X,Y,Z \) range over propositional variables and \( a,b,c,d, \ldots \) over propositional constants.

MALL2 formulas \( F \) are built up recursively from \( \mathcal{V} \) and \( \mathcal{C} \) by the following grammar:

\[
F ::= X | a | X^⊥ | a^⊥ | 1 | \bot | T | 0 | F \otimes F | F \otimes F \otimes F | F \land F | F \lor F | F \lor F \lor F | \forall XF | \exists XF.
\]

As usual, the negation of a MALL2 formula is defined recursively by using involutivity of negation and the de Morgan laws.

General formulas will be referenced by the capital letters \( F, G, H \) and atomic formulas by the capital letters \( A, B, C \). Multi-sets of formulas will be referenced by the Greek letters \( \Delta, \Gamma \).

The inference rules for MALL2 are given in Fig. 1. In the application of the \( \exists \)-rule, we use the notation \( \overline{X} \) of Emms [2] for the formula \( G \) that instantiates the quantified
identity group
\[
\vdash A, A \text{id} \quad \vdash A_1, F \vdash F, A_2 \text{cut}
\]

logical group

\[\text{multiplicatives}\]
\[
\vdash A_1, F_1 \vdash F_2, A_2 \quad \vdash F_1, F_2, A \text{\&} \\
\vdash A_1, F_1 \& F_2, A_2 \quad \vdash F_1 \& F_2, A
\]

\[\text{additives}\]
\[
\vdash F_1, A \vdash F_2, A \quad \vdash F_1, A \text{\&} F_2, A \quad \vdash F_1 \oplus F_2, A \text{\&} \\
\vdash F_1 \oplus F_2, A \quad \vdash F_1, A \oplus F_2, A \quad \vdash F_1, A \oplus F_2, A \text{\&}
\]

\[\text{second-order quantifiers}\]
\[
\vdash F[X,Y], A \forall Y \quad \vdash F[X,Y], A \exists Y \\
\vdash \forall X F, A \quad \vdash \exists X F, A \\
\text{with } Y \text{ not free in the conclusion}
\]

Fig. 1. The rules of MALL2 sequent calculus.

variable \(X\) when this formula is not determined immediately. \(X\) is called an unknown formula.

Before we get to the heart of the matter, we need to define the vocabulary related to bottom-up proof search in MALL2 which we use in the rest of the article. In fact, this vocabulary is not especially related to MALL2 and it can apply to any deductive system that is formalised in the sequent calculus.

Definition 1.1. A goal is a finite set of MALL2 sequents, which are its subgoals. A deduction is a finite, or infinite, sequence of goals (\(G_n\)) such that for any \(n, G_{n+1}\) is obtained from \(G_n\) by replacing a part \(\{\vdash A_1, \ldots, \vdash A_q\}\) of \(G_n\) (its active subgoals) with a finite (possibly empty) set of new subgoals \(\{\vdash A'_1, \ldots, \vdash A'_q\}\) such that \(\vdash A_1, \ldots, \vdash A_q\) are derivable from \(\vdash A'_1, \ldots, \vdash A'_q\) in MALL2.

A deduction is successful if it is finite and its last goal is empty.

Usually, the initial goal \(G_0\) of a deduction contains only one subgoal, the sequent to prove. Usually too, in a deduction step there is one active subgoal but in some cases, as we will see later with the principle of elimination of unknowns, several subgoals can be active at the same time.
Table 1
Inference permutability in MALL2

<table>
<thead>
<tr>
<th>( t_2 \setminus t_1 )</th>
<th>cut</th>
<th>( \otimes )</th>
<th>( \exists )</th>
<th>( \perp )</th>
<th>&amp;</th>
<th>( \oplus )</th>
<th>( \forall )</th>
<th>( \exists )</th>
</tr>
</thead>
<tbody>
<tr>
<td>cut</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \otimes )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \exists )</td>
<td>np</td>
<td>np</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \perp )</td>
<td>np</td>
<td>np</td>
<td>np</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&amp;</td>
<td>np</td>
<td>np</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \oplus )</td>
<td>np</td>
<td>np</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \forall )</td>
<td>np</td>
<td>np</td>
<td>np</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>( \exists )</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Definition 1.2.** A deduction procedure is a set of deductions.

A deduction procedure is complete if, for any provable sequent \( \Gamma \vdash \Delta \), there exists a successful deduction that belongs to this procedure and that begins with the goal \( \{ \Gamma \vdash \Delta \} \). A deduction procedure terminates if, for any sequent \( \Gamma \vdash \Delta \), all deductions that belong to this procedure and that begin with the goal \( \{ \Gamma \vdash \Delta \} \) are finite and their number is also finite.

A deduction procedure is a decision procedure if it is complete and it terminates.

The most general deduction procedure in MALL2, which we denote by \( \mathcal{P}_0 \), consists in deductions where, at each step, we apply a rule of the MALL2 sequent calculus from the conclusion to the premises. We drop the cut-rule from \( \mathcal{P}_0 \) and from all deduction procedures because it is highly non-deterministic in bottom-up proof search. Despite this restriction, because of the redundancy of the cut-rule, \( \mathcal{P}_0 \) remains complete but obviously it does not terminate. Our aim is to restrict \( \mathcal{P}_0 \) as much as possible while keeping completeness. As MALL2 is undecidable, we cannot hope to obtain a procedure for the whole of MALL2 which terminates. A way of restricting the procedure is to start from the following observation: because we use the formalism of the sequent calculus, a substantial part of the order between the rules that are applied during a deduction is irrelevant and so it can be fixed so that non-determinism of the procedure is reduced. To distinguish what is relevant and what is irrelevant in the order of applied rules, we have to make a static analysis of inference permutability in MALL2 proofs. Then we transpose the conclusions of such an analysis into the definition of deduction procedures.

1.2. Inference permutability in MALL2

The property of inference permutability in a proof means the possibility of two consecutive inferences inverting without disturbing the rest of the proof. This property is defined precisely in [5] where it is studied for first order linear logic. The results that conclude this study can be easily transposed to second order linear logic. Here, we are only interested in the fragment MALL2 for which inference permutability can be summarised in Table 1. There are eight cases of non-permutability corresponding to the eight entries of the table that are marked by \( np \).\(^1\)

\(^1\) For more details about the meaning of the table, see [5].
If we analyse this table column by column, we find that the columns with $\&$, $\bot$, $\not\exists$ and $\lor$ at the head are empty. That means that the corresponding inferences can be moved downwards in a proof as far as possible in the sense given in [5].

Now, if we analyse this table line by line, we find that the lines with $\otimes$, $\bot$, $\not\exists$ and $cut$ at the head are empty. That means that the corresponding inferences can be moved upwards in a proof as far as possible.

As a consequence, we can divide the inference types into two classes $T \downarrow$ and $T \uparrow$ according to the direction in which they can move most easily in a proof.

$$T \downarrow = \{\&, \bot, \not\exists, \lor\} \quad \text{and} \quad T \uparrow = \{\otimes, \not\exists, \exists, cut\}$$

We have chosen to put inferences of type $\bot$ in $T \downarrow$ because here we consider bottom-up proof construction and our interest is to delete the constants $\bot$ as soon as they emerge in a subgoal.

Using this, we normalise proofs by moving their inferences of a type belonging to $T \downarrow$ as far downwards and moving their inferences of a type belonging to $T \uparrow$ as far upwards as possible. This normalisation includes cut elimination but it goes further in ordering inferences in a particular way as the following example shows.

**Example 1.1.** Here is the beginning of a proof $\Pi$ in MALL2.

\[
\vdash \not\exists a, a, a \otimes b \quad \text{id}
\]
\[
\vdash \not\exists a, a, a \otimes b \quad \vdash b, b
\]
\[
\vdash \not\exists a, a \otimes b, a \otimes b, b
\]
\[
\vdash \not\exists a, a \otimes b \vdash (\not\exists a) \not\exists (a \otimes b), a \otimes b, b
\]
\[
\vdash \not\exists a, a \otimes b \vdash (\not\exists a) \not\exists (a \otimes b), a \otimes b, b
\]
\[
\vdash \not\exists a, a \otimes b \vdash (\not\exists a) \not\exists (a \otimes b), a \otimes b, b
\]
\[
\vdash \not\exists a, a \otimes b \vdash (\not\exists a) \not\exists (a \otimes b), a \otimes b, b
\]
\[
\vdash \exists x (x \otimes ((\not\exists a) \not\exists (a \otimes b)), a \otimes b, b)
\]

This proof is not normal in the sense given above because we can permute the last inference of type $\not\exists$ with the inference of type $\exists$ and we can also permute the first inference of type $\not\exists$ with the first inference of type $\otimes$. In this way, we obtain the
following proof $\Pi_n$:

\[
\vdash \overrightarrow{X}, \ a, \ a \otimes b \quad \vdash b, \ b
\]

\[
\vdash \overrightarrow{X}, \ a \otimes b, \ a \otimes b, \ b
\]

\[
\vdash \overrightarrow{X}, \ a \otimes (\overrightarrow{a} \otimes b), \ a \otimes b, \ b
\]

\[
\vdash \exists X (X \otimes (X \overrightarrow{a} \otimes b), \ a \otimes b, \ a \otimes b, \ b)
\]

If we continue to re-order inferences in $\Pi_n$, we obtain a normal proof.

As the resulting class of normal proofs is complete, we can restrict the proof search to normal proofs in order to establish the provability of sequents. This restricts deduction procedures in two ways:

- As soon as a formula of a type in $T \downarrow$ emerges in a subgoal, we decompose it. We have described this principle in [4] as the principle of immediate decomposition. This corresponds to the fact that formulas of a type in $T \downarrow$ are introduced as low as possible in normal proofs. Andreoli expresses this in [1] with the notion of invertibility.

- As soon as a formula of a type in $T \uparrow$ begins to be decomposed, the decomposition goes on as long as the components are of a type in $T \uparrow$. We have described this principle in [4] as the principle of chaining decomposition. This corresponds to the fact that formulas of a type in $T \uparrow$ are introduced as high as possible in normal proofs. This corresponds to the notion of focusing introduced in [1].

We can embed both principles into the procedure $\mathcal{P}_0$ in a formal way to obtain the procedure $\mathcal{P}_1$.

**Definition 1.3.** The procedure $\mathcal{P}_1$ is the set of deductions $(G_n)$ belonging to $\mathcal{P}_0$ such that every deduction step $G_n \rightarrow G_{n+1}$ has the following properties:

- if a subgoal of $G_n$ contains a formula of a type in $T \downarrow$, then $G_n \rightarrow G_{n+1}$ consists in decomposing such a formula (immediate decomposition principle);
Example 1.2. The first proof \( \Pi \) of Example 1.1 does not result from a deduction that belongs to \( \mathcal{P}_1 \) because it begins with an application of the \( \exists \)-rule whereas the initial sequent contains a formula \((a^\perp \otimes b^\perp) \exists b^\perp \) of a type in \( T \downarrow \).

The second proof \( \Pi_n \) of Example 1.1 results from a deduction that belongs to \( \mathcal{P}_1 \): this is an alternation of immediate decomposition and chaining decomposition phases and each chaining decomposition phase is superposed with the immediate decomposition phase that follows by means of its last step. Here, we have successively:

- an immediate decomposition phase (an application of the \( \exists \)-rule);
- a chaining decomposition phase (applications of the \( \exists, \otimes \) and \( \exists \) rules);
- an immediate decomposition phase (two applications of the \( \exists \)-rule).

The interest of the procedure \( \mathcal{P}_1 \) is that it reduces non-determinism in proof search while preserving completeness [4]. Unfortunately, \( \mathcal{P}_1 \) does not terminate in MALL2.

2. A decision procedure for a fragment of MALL2

The \( \exists \)-rule is the source of non-termination for every deduction procedure that aims to be complete for all sequents of MALL2 as is shown in [6]. The problem stems from the fact that, in a deduction, a formula of the form \( \exists X F \) can be decomposed into \( F[G/X] \) where all occurrences of \( X \) are replaced by an arbitrary formula \( G \).

In one case, this difficulty can be easily overcome: when all occurrences of \( X \) are either positive or negative. If they are positive, \( X \) can be instantiated by the constant \( \top \) and if they are negative, \( X \) can be instantiated by the constant \( 0 \). Then, all subgoals where the formula \( X \) surfaces, will be satisfied automatically by means of the \( \top \)-axiom.

2.1. The principle of elimination of unknowns

In the general case, we try to postpone the instantiation of \( X \) as late as possible until we have enough information to do it. In this case, this means when all occurrences of the unknown formula \( X \) surface in the subgoals. At this moment, we eliminate \( X \) by means of the following principle.

Elimination of unknowns. In a deduction, when \( X \) is only present in subgoals that have the form \( \vdash X, A_1 \cdots \vdash X, A_n \vdash X^\perp, A'_1 \cdots \vdash X^\perp, A'_m \), we replace these subgoals by the subgoals \( \vdash A_i, A'_j \) where \((i, j)\) is any element of \([1, n] \times [1, m] \).
As we shall see in the proof of the lemma below, that amounts to giving the value \( \mathfrak{S}(A'_1) \& \cdots \& \mathfrak{S}(A'_m) \) to \( \overline{X} \). This is also equivalent to giving it the value \( (\mathfrak{S}(A_1))^1 \oplus \cdots \oplus (\mathfrak{S}(A_n))^\bot \).

The principle of elimination of unknowns is justified by the following lemma which is a generalisation of the lemma of elimination of unknowns introduced by Emms in [2].

**Lemma 2.1.** There exists a MALL2 formula \( F \) such that the sequents \( \vdash F, A_1 \cdots \vdash F, A_n \vdash F^\bot, A'_1 \cdots \vdash F^\bot, A'_m \) are provable if and only if the sequents \( \vdash A_i, A'_j \) where \((i,j)\) is any element of \([1,n] \times [1,m]\), are provable.

**Proof 2.1.** The left to right direction follows immediately from the cut rule. For the right-to-left direction, we choose \( F \) to be the formula \( \mathfrak{S}(A'_1) \& \cdots \& \mathfrak{S}(A'_m) \). The provability of \( \vdash F, A_i \) follows from \( m-1 \) applications of the \&-rule. The formula \( F^\bot \) has the form \( \mathfrak{S}(A'_1)^\bot \oplus \cdots \oplus \mathfrak{S}(A'_m)^\bot \). To prove \( \vdash F^\bot, A'_j \) we start from the provable sequent \( \vdash \mathfrak{S}(A'_j)^\bot, A'_j \) and we apply the rules \( \oplus_1 \) or \( \oplus_2 \) \( m-1 \) times. \(\square\)

After decomposition of an existentially quantified formula, the resulting goal must have the appropriate form for the principle of elimination of unknowns to be applied: all generated subgoals must contain one occurrence of \( \overline{X} \) or \( \overline{X}^\bot \) at most. This is not always the case as the following example shows.

**Example 2.1.** Let us consider the formula \( F = \forall X((X \& 1) \rightarrow (X \otimes X)) \) used by Y. Lafont for simulating contraction in MALL2 [6]. As \( F \) is present on the left-hand side of sequents, it will be translated into its negation \( F^\bot = \exists X((X \& 1) \otimes (X^\bot \mathfrak{S}X^\bot)) \) in one-sided sequents. Let us decompose the formula \( F^\bot \) in a deduction. We obtain the following derivation tree:

\[
\begin{align*}
\vdash \overline{X}, A_1 \vdash 1, A_1 & \quad \vdash \overline{X}^\bot, \overline{X}^\bot, A_2 \\
\hline
\vdash \overline{X} \& 1, A_1 & \quad \vdash \overline{X}^\bot \mathfrak{S} \overline{X}^\bot, A_2 \\
\hline
\vdash (\overline{X} \& 1) \otimes (\overline{X}^\bot \mathfrak{S} \overline{X}^\bot), A_1, A_2 \\
\hline
\vdash \exists X((X \& 1) \otimes (X^\bot \mathfrak{S}X^\bot)), A_1, A_2
\end{align*}
\]

In the deduction, the initial goal is replaced with three subgoals but the presence of two occurrences of \( \overline{X}^\bot \) in one of these prevents the application of the principle of elimination of unknowns. The reason for this presence is that the subformula \( X^\bot \mathfrak{S}X^\bot \) of \( F \) contains the quantified variable \( X \) in both of its components.

---

3 \( \mathfrak{S}(A) \) represents the unique formula that is obtained by linking all formulas of \( A \) with the connective \( \mathfrak{S} \).
We could imagine an extension of the principle of elimination of unknowns to cases where sequents contain more than one occurrence of the unknown. By means of such an extended principle, we could replace the two subgoals $\vdash \overline{X}, \Delta_1$ and $\vdash \overline{X}^\perp, \overline{X}^\perp, \Delta_2$ with the unique subgoal $\vdash \Delta_1, \Delta_1, \Delta_2$ in the previous example. In doing this, we duplicate the multi-set $\Delta_1$ with respect to the initial subgoal. This is a problem because we aim to have a deduction procedure that terminates, which can be simply guaranteed by the size decrease of the subgoals. For this reason, we keep the principle of elimination of unknowns in its initial form.

Now, if we want to be able to always apply it after the decomposition of a formula $\exists X F$, a sufficient condition is that two occurrences of $X$ do not belong to two distinct components of a subformula of $F$ of type $\forall$. Henceforth, we only consider existential formulas with the following syntactical restriction.

**Condition 1.** If a formula $\exists X F$ contains occurrences of both $X$ and $X^\perp$, and if $F_1 \forall F_2$ is a subformula of $F$, then $F_1$ and $F_2$ contain no occurrences of $X$ or $X^\perp$ at the same time.

This condition is not sufficient to guarantee the strict size decrease of the subgoals. In the application of the principle of elimination of unknowns, we generate sub-goals in the form $\vdash \Delta_i, \Delta_j$ which can be greater than the unique subgoal $\vdash \exists X F, \Delta$ from which they come. They can be greater because, between the decomposition of $F$ and the elimination of the unknown $\overline{X}$, an application of the $\&$-rule gives rise to a duplication of context. The following example illustrates this problem.

**Example 2.2.** Let us consider the formula $\exists X ((X \otimes a) \oplus (X^\perp \otimes b)) \forall (a^\perp \& b^\perp)$. We call this formula $F$ and we aim to prove the goal $\vdash F, F$. For this, let us suppose that we build the following derivation:

$$
\begin{array}{c}
\vdash \overline{X}, F \vdash a, \overline{a} \\
\hline
\vdash X \otimes a, \overline{a}, F
\end{array}
\text{id}
$$

$$
\begin{array}{c}
\vdash \overline{X}^\perp, F \vdash b, \overline{b} \\
\hline
\vdash \overline{X}^\perp \otimes b, b^\perp, F
\end{array}
\text{id}
$$

$$
\begin{array}{c}
\vdash (\overline{X} \otimes a) \oplus (\overline{X}^\perp \otimes b), a^\perp, F \\
\hline
\vdash (\overline{X} \otimes a) \oplus (\overline{X}^\perp \otimes b), b^\perp, F
\end{array}
\end{array}
\oplus_1
$$

$$
\begin{array}{c}
\vdash (\overline{X} \otimes a) \oplus (\overline{X}^\perp \otimes b), a^\perp \& b^\perp, F \\
\hline
\vdash ((\overline{X} \otimes a) \oplus (\overline{X}^\perp \otimes b)) \forall (a^\perp \& b^\perp), F
\end{array}
\oplus_2
$$

$$
\begin{array}{c}
\vdash F, F
\end{array}
\&
$$
Then, we apply the principle of elimination of unknowns to \( \vdash X, F \) and \( \vdash X^\perp, F \) and we recover the initial goal from these subgoals. The size of the goal does not decrease and the procedure loops although \( \vdash F, F \) is provable.

The application of the principle of elimination of unknowns is the source of another problem. By postponing the instantiation of the variable for the \( \exists \)-rule, we can violate the side condition of the \( \forall \)-rule which could have been applied in the meantime. Here is an example of such a violation.

**Example 2.3.** Let us consider the goal \( \vdash \exists \forall Y ((X \forall Y) \otimes (X^\perp \forall Y)) \). By decomposing the formula, we obtain the following derivation tree:

\[
\begin{array}{c}
\vdash \overline{X}, \overline{Z} \\
\vdash \overline{X}^\perp, \overline{Z} \\
\vdash \overline{X} \forall \overline{Z} \\
\vdash \overline{X}^\perp \forall \overline{Z} \\
\vdash ((\overline{X} \forall \overline{Z}) \otimes (\overline{X}^\perp \forall \overline{Z})) \\
\vdash \forall Y ((\overline{X} \forall Y) \otimes (\overline{X}^\perp \forall Y)) \\
\vdash \exists \forall Y ((\overline{X} \forall Y) \otimes (\overline{X}^\perp \forall Y))
\end{array}
\]

Then, by elimination of the unknown \( \overline{X} \), we replace \( \vdash \overline{X}, \overline{Z} \) and \( \vdash \overline{X}^\perp, \overline{Z} \) by the unique subgoal \( \vdash Z, Z^\perp \), which is provable. However, the initial sequent \( \vdash \exists \forall Y ((Y \forall X) \otimes (X^\perp \forall Y)) \) is not provable. The problem comes from the fact that the elimination of unknowns corresponds to the instantiation of \( X \) by the fresh variable \( Z \), which constitutes a violation of the condition governing the \( \forall \)-rule: \( Z \) must not be free in the conclusion of the corresponding inference.

We can hope to prevent such violations by marking each unknown \( \overline{X} \) with the fresh variables that appear during its existence. Then, in the step of elimination of \( \overline{X} \), it remains to verify that the formula \( \forall (A'_1) \& \cdots \& \forall (A'_n) \) which instantiates \( X \) does not contain marked variables.

Since this last problem can be solved easily, let us come back to the first problem, the non-termination of deductions because of a bad combination of the \&-rule and the elimination of unknowns. When dealing with general deductions, we cannot easily find syntactical conditions that guarantee their termination as Example 2.2 illustrates. The idea is to restrict the syntax of the existentially quantified formulas in such a way that we can apply the following principle which simplifies the search for such conditions of termination.

**One-piece decomposition.** In a deduction, as soon as we start to decompose a formula \( \exists X F \), we go on with the decomposition until all occurrences of \( \overline{X} \) surface in the goal.
2.2. One-piece decomposition of existentially quantified formulas

The one-piece decomposition principle is interesting insofar as it preserves the completeness of the deduction procedure. This holds if it consists in applying the chaining decomposition principle and then the immediate decomposition principle. For this, a formula that is existentially quantified must have a particular syntax: it must be decomposable at one go.

Definition 2.1. A formula $\exists X F$ is decomposable at one go if each occurrence $L$ of $X$ or $X^\perp$ in $F$ has the following property: if $L$ belongs to a subformula $F'$ of $F$ with the type $\&$, $\lor$ or $\forall$, no subformula of $F'$ that contains $L$ has the type $\exists$, $\oplus$ or $\ominus$.

We can decompose such a formula $\exists X F$ at one go to make all occurrences of $X$ emerge because we can begin by applying rules of $T^\uparrow$ and using the chaining decomposition principle. Then, we can terminate by applying rules of $T^\downarrow$ and using the immediate decomposition principle.

Example 2.4. Let us consider the normal proof $\Pi_n$ of Example 1.1 again.

\[
\vdash X^\perp, a^\perp, a^\perp \otimes b^\perp \quad \vdash b, b^\perp
\]

\[
\vdash X^\perp, a^\perp, a \otimes b, a^\perp \otimes b^\perp, b^\perp
\]

\[
\vdash \exists (X^\perp \& a^\perp) \& (a \otimes b), a^\perp \otimes b^\perp, b^\perp
\]

\[
\vdash (X \otimes ((X^\perp \& a^\perp) \& (a \otimes b))), a \otimes b, a^\perp \otimes b^\perp, b^\perp
\]

\[
\vdash \exists (X \otimes ((X^\perp \& a^\perp) \& (a \otimes b))), a \otimes b, (a^\perp \otimes b^\perp), \& b^\perp
\]

In $\Pi_n$, the formula $F = \exists (X \otimes ((X^\perp \& a^\perp) \& (a \otimes b)))$ is decomposed at one go: first, the application of the $\exists$, $\otimes$ and $\&$ rules constitutes a phase of chaining decomposition; the application of the $\&$-rule marks its end and the beginning of an immediate decomposition phase; this phase is constituted of two applications of the $\&$-rule which achieves the decomposition of $F$. Then, instead of continuing with the application of the $\otimes$-rule, we can apply the principle of elimination of unknowns to terminate the proof. Here, $F$ is decomposable at one go because its syntax corresponds to Definition 2.1: the connectives $\otimes$ and $\&$ are in correct order in $F$. 
Now, let us consider the following derivation:

\[
\begin{align*}
\vdash \overline{X}, b & \vdash a, a^\perp \\
\vdash \overline{X}, \otimes a, a^\perp, b & \otimes \\
\vdash \overline{X} \otimes a, a^\perp \mathcal{B} b & \vdash \overline{X}, b^\perp \\
\vdash (\overline{X} \otimes a) \mathcal{B} \overline{X}, (a^\perp \mathcal{B} b) \otimes b^\perp & \exists \\
\vdash \exists X((X \otimes a) \mathcal{B} X^\perp), (a^\perp \mathcal{B} b) \otimes b^\perp
\end{align*}
\]

In this derivation, the formula \( G = \exists X((X \otimes a) \mathcal{B} X^\perp) \) is not decomposed at one go and this is not possible because of the order between the connectives \( \otimes \) and \( \mathcal{B} \) in the syntax of \( G \). Nevertheless, even if \( G \) is not decomposable at one go and does not verify Condition 1, we can eliminate the unknown \( \overline{X} \) to terminate the proof.

Henceforth, we only consider existential formulas with the following syntactical restriction.

**Condition 2.** If a formula \( \exists X F \) contains occurrences of both \( X \) and \( X^\perp \), \( \exists X F \) is decomposable at one go.

In a deduction, after application of the one-piece decomposition principle to such formulas and because of Condition 1, the resulting goal has the appropriate form so that the principle of elimination of unknowns can be applied since all generated subgoals contain one occurrence of \( \overline{X} \) or \( \overline{X}^\perp \) at most. Now, we have to make sure that the procedure terminates.

\[ 2.2.1. \text{Termination conditions} \]

If we choose the number of connectives as a measure of the size of the sequents, we notice that at every step of a deduction, the size of the active subgoal is strictly greater than the size of each subgoal that replaces it except when we apply the principle of elimination of unknowns. In this case, we compare the size of each generated subgoal \( \vdash A_i, A_j \) with the size of the unique subgoal \( \vdash \exists X F \), \( A \) from which they come. It can be greater because, in the decomposition of \( F \), an application of the \( \& \)-rule gives rise to a duplication of context.

**Example 2.5.** Let us consider the goal \( \vdash \exists X(X \& X^\perp), \exists X(X \& X^\perp) \). By application of the one-piece decomposition principle, we replace it with two subgoals \( \vdash \overline{X}, \exists X(X \& X^\perp) \) and \( \vdash \overline{X}, \exists X(X \& X^\perp) \). Then, we apply the principle of elimination of unknowns
and we recover the initial goal from these subgoals. The size of the goal does not decrease and the deduction loops.\footnote{By projecting the sequent $\vdash \exists X (X \& \neg X)$, $\forall X (X \& \neg X)$ in classical logic, we obtain the non provable sequent $\vdash \exists X (X \& \neg X)$, $\exists X (X \& \neg X)$, therefore the initial sequent is not provable in $\text{MALL}_2$.}

To avoid this problem, we restrict the syntax of each formula $\exists X F$ as follows.

**Condition 3.** If a formula $\exists X F$ contains occurrences of both $X$ and $X^\bot$, then $X$ and $X^\bot$ do not belong to distinct components of a subformula of $F$ of type $\&$.

2.2.2. Completeness conditions

As the instantiation of an existentially quantified variable $X$ is delayed until the phase of elimination of unknowns, this instantiation can violate the side condition of the $\forall$-rule which could have been applied during the phase of one-piece decomposition. By marking the unknown with the fresh variables that appear, we can prevent this violation but at the same time we lose the completeness of the procedure as the following example shows.

**Example 2.6.** We want to prove the sequent $\vdash \exists X ((X \& a^\bot) \& Y (X^\bot \& (a \oplus Y)))$. First, we apply the principle of one-piece decomposition and in this phase, we mark $X$ with the fresh variables that appear during its existence. We obtain the following derivation tree:

\[
\begin{array}{c}
\vdash X^\bot, a \oplus Z \\
\vdash X^\bot, a \\
\vdash X^\bot, a \oplus Z \\
\vdash X^\bot, a \oplus Z \\
\vdash (X^\bot, a \oplus Z) \& Y (X^\bot, a \oplus Z) \\
\vdash \exists X ((X \& a^\bot) \& Y (X^\bot \& (a \oplus Y)))
\end{array}
\]

Then the phase of elimination of unknowns consists in replacing the subgoals $\vdash X$, $a^\bot$ and $\vdash X^\bot$, $a \oplus Z$ with $\vdash a^\bot$, $a \oplus Z$, which corresponds to the instantiation of $X$ with $a \oplus Z$. According to the mark $Z$ of $X$, this constitutes a violation of the side condition related to the $\forall$-rule and the deduction fails. Nevertheless, the initial sequent is provable because there is another instantiation of $X$ with the constant "a" that is sound and works but this is not generated automatically by the procedure.

We avoid the presence of fresh variables in the formula that instantiates the unknown $X$ in the phase of elimination of unknowns by constraining the syntax of formulas $\exists X F$ as follows.
Condition 4. If a formula $\exists X F$ contains occurrences of both $X$ and $X^\perp$ and if $\forall Y G$ is a subformula of $F$, $X$ and $Y$ do not belong to two distinct components of a subformula of $G$ of type $\exists$.

2.2.3. Interferences between applications of the $\exists$-rule

Until now, we have implicitly assumed that the one-piece decomposition of a formula $\exists X F$ does not entail the decomposition of a subformula of $F$ which has type $\exists$ also. However, such a case may occur but causes two problems. The first problem comes from the fact that the formula $\exists X F$ being decomposed at one go can contradict the decomposition at one go of one of its subformulas.

Example 2.7. Let us consider the sequent $\vdash \exists X \exists Y (X \otimes Y \otimes ((X^\perp \mathcal{R} a) \mathcal{R} (Y^\perp \mathcal{R} b))), \Delta$. The first formula of the sequent is decomposable at one go. Then, trying to apply the principle of one-piece decomposition, we obtain the following derivation tree:

$$
\vdash \overline{X}^\perp, \overline{Y}^\perp, a, b, A_3
$$

$$
\vdash \overline{X}^\perp, \overline{Y}^\perp, \mathcal{R} b, a, A_3
$$

$$
\vdash \overline{X}^\perp, A_1 \vdash \overline{Y}^\perp, A_2 \otimes
\vdash \overline{X}^\perp \mathcal{R} a, \overline{Y}^\perp \mathcal{R} b, A_3
$$

$$
\vdash \overline{X}^\perp \mathcal{R} a \mathcal{R} (Y^\perp \mathcal{R} b), A_3
$$

$$
\vdash \exists Y (\overline{X} \otimes Y \otimes ((\overline{X}^\perp \mathcal{R} a) \mathcal{R} (\overline{Y}^\perp \mathcal{R} b)), \Delta
$$

At the end of the phase, the initial subgoal is replaced by three new subgoals and now we have two unknowns to eliminate. We achieve this by applying the principle of elimination of unknowns twice and we obtain the subgoal $\vdash a, b, A_1, A_2, A_3$.

Now, if we examine the derivation carefully, we observe that, if the formula $\exists X \exists Y (X \otimes Y \otimes ((X^\perp \mathcal{R} a) \mathcal{R} (Y^\perp \mathcal{R} b)))$ is decomposed at one go with respect to $X$. This is not the case for the subformula $\exists Y (X \otimes Y \otimes ((X^\perp \mathcal{R} a) \mathcal{R} (Y^\perp \mathcal{R} b))$ with respect to $Y$ because the two applications of the one piece decomposition principle are mutually exclusive.

The second problem is more embarrassing because it prevents the elimination of some unknowns by application of our principle. It comes from the possibility for two different unknowns being present in two different formulas of the same subgoal.
Example 2.8. Let us consider the sequent \( \vdash \exists X \exists Y ((X \forall Y) \otimes (X^\perp \forall Y^\perp)) \). By decomposing its unique formula at one go, we obtain the following derivation:

\[
\begin{align*}
\vdash X, \bar{Y} & \quad \vdash X^\perp, \bar{Y}^\perp \\
\hline & \vdash X \forall \bar{Y} \quad \vdash X^\perp \forall \bar{Y}^\perp \\
\hline & \vdash (X \forall \bar{Y}) \otimes (X^\perp \forall \bar{Y}^\perp) \quad \exists \\
\hline & \vdash \exists Y((X \forall \bar{Y}) \otimes (X^\perp \forall \bar{Y}^\perp)) \quad \exists \\
\hline & \vdash \exists X \exists Y((X \forall \bar{Y}) \otimes (X^\perp \forall Y^\perp))
\end{align*}
\]

After elimination of the unknown \( \bar{X} \), we obtain the sequent \( \vdash \bar{Y}, \bar{Y}^\perp \). This sequent is provable but our deduction procedure halts because we have two instances of the same unknown in the sequent to prove and we cannot apply the principle of elimination of unknowns.

A way of avoiding both problems is to restrict the syntax of existentially quantified formulas much more by rewriting Condition 1.

**Condition 1 (revised version).** If a formula \( \exists X F \) contains occurrences of both \( X \) and \( X^\perp \), if \( F_1 \forall F_2 \) is a subformula of \( F \), and if \( F_1 (F_2) \) contains an occurrence of \( X \) or \( X^\perp \), then \( F_2 (F_1) \) has the following property: for any subformula \( \exists Y G \) of \( \exists X F \) which contains occurrences of both \( Y \) and \( Y^\perp \), \( Y \) is not free in \( F_2 (F_1) \).

The revised condition guarantees that, at each step of a deduction, there is one formula with at most one unknown in each subgoal. So, the source of the second problem disappears and the first problem is essentially solved: there remains a possible contradiction between two decompositions at one go which is due to the fact that the subgoals of a same goal are proved sequentially and not concurrently. This requires a slight relaxation of the one piece decomposition principle, which can be formulated as follows.

**One-piece decomposition (revised version).** In a deduction, if a goal contains a complex formula with an unknown, then the next deduction step consists in decomposing such a formula.

With this formulation, several existentially quantified formulas can be decomposed within a phase of one piece decomposition without problems.

2.3. The PSPACE-complete fragment of \( \text{MALL2} \)

Now, we are in a position to make precise the syntax of the fragment of \( \text{MALL2} \) for which it will be possible to define a decision procedure. This fragment is denoted \( \text{MALL2}' \).
Definition 2.2. A MALL2' formula is a MALL2 formula $F$ such that all existentially quantified subformulas of $F$ verify Conditions 1–4.

Example 2.9. The formulas $\exists X((X \& 1) \otimes (X \mid \& X \mid))$, $\exists X(X \& X \mid)$ and $\exists X \forall Y((X \otimes Y \mid) \otimes (X \mid \& Y \mid))$ are not MALL2' formulas respectively because of conditions 2, 3 and 4 in the definition of MALL2'.

On the other hand, $\exists X((X \mid \& a) \otimes \forall Y(Y \& X))$ and $\exists X \exists Y(X \otimes Y \otimes (X \mid \& Y \mid))$ are MALL2' formulas.

The determination of the membership of MALL2' for a MALL2 formula is linear time: it consists in exploring just once the syntactic tree of the formula from the leaves to the root.

Now, we have to exhibit a decision procedure for MALL2'. This procedure, called $\mathcal{P}_2$, essentially consists in restricting the application of the $\exists$-rule with the principles of one-piece decomposition and elimination of unknowns.

Definition 2.3. The procedure $\mathcal{P}_2$ of MALL2' is a set of deductions $(G_n)$ such that any deduction step $G_n \rightarrow G_{n+1}$ verifies the following properties:

(a) If $G_n$ does not contain any unknown formula, $G_n$ is deduced from $G_{n+1}$ by application of a MALL2 inference rule. When the $\exists$-rule is applied to decompose a formula $\exists X F$ such that $X$ is free in $F$, three cases are possible:
   1. $F$ contains only positive occurrences of $X$ and $X$ is instantiated with the constant $T$;
   2. $F$ contains only negative occurrences of $X$ and $X$ is instantiated with the constant 0;
   3. $F$ contains both positive and negative occurrences of $X$ and $X$ is not instantiated immediately but it is replaced by the unknown $\overline{X}$.

(b) If $G_n$ contains a complex formula with an occurrence of an unknown formula, the step $G_n \rightarrow G_{n+1}$ consists in applying the one-piece decomposition principle;

(c) If $G_n$ contain an unknown formula and if all unknowns of $G_n$ are un-nested, then one of these unknowns, $\overline{X}$, is eliminated according to the principle of elimination of unknowns.

Now, we come to the central theorem, which is a natural consequence of the four conditions that define the syntax of MALL2' formulas. Its proof requires two lemmas. The first gives a characteristic of the deductions that belong to $\mathcal{P}_2$.

Lemma 2.2. In each subgoal of a goal that belongs to a deduction of $\mathcal{P}_2$, there is at most one formula which contains unknowns and this formula also contains all fresh variables that have been introduced after one of the unknowns.

Proof 2.2. Let $(G_n)$ be any deduction of $\mathcal{P}_2$. We prove by induction on the rank $n$ of the step in the deduction that any goal $G_n$ has property Prop which is defined in the lemma.

The goal $G_0$ has property Prop because its subgoals contain no unknowns.
We prove the induction step from \( G_n \) to \( G_{n+1} \) by considering the three cases according to Definition 2.3.

**case (a)** \( G_n \) contain no unknown and thus \( G_{n+1} \) contain one formula with at most one unknown when the \( \exists \)-rule is applied. Therefore, \( G_{n+1} \) verifies property \( \Pr \).

**case (b)** By induction hypothesis, the goal \( G_n \) verifies \( \Pr \). If the rule that is applied from \( G_n \) to \( G_{n+1} \) is the \( \otimes, \&_1 \) or \( \oplus_2 \)-rule, the goal \( G_{n+1} \) also verifies \( \Pr \) because, in the new subgoals, there is only one formula which comes from the decomposition of the main formula of the active subgoal.

If the rule that is applied is the \( \forall \) or \( \exists \)-rule, the formula that contains the unknowns and the possible concerned fresh variable is the main formula of the active subgoal because we apply the principle of decomposition at one go and thus, the goal \( G_{n+1} \) has property \( \Pr \).

If the rule that is applied is the \( \exists \)-rule, the main formula is decomposed into two formulas in the same subgoal but, because of Conditions 1 and 4, only one contains unknowns and fresh variables which have been introduced after these unknowns, thus the goal \( G_{n+1} \) has property \( \Pr \).

**case (c)** By induction hypothesis, all active subgoals contain no critical fresh variables and the unique unknown which they contain, either remains or is eliminated in \( G_{n+1} \). Therefore, \( G_{n+1} \) has property \( \Pr \). \( \square \)

The second lemma fixes some bounds in the size of goals that belong to a deduction of \( \mathcal{P}_2 \).

**Lemma 2.3.** Let \( \vdash \exists X F, \Lambda \) be a sequent that is decomposed in a phase of decomposition at one go using \( \mathcal{P}_2 \). The current goal \( G \) at each step of such a phase has the following property, which we call \( \Pr \):

1. each subgoal of \( G \) that contains positive and negative occurrences of an unknown has size strictly less than \( \vdash \exists X F, \Lambda \);
2. if a subgoal \( \vdash \Delta_1 \) of \( G \) contains a positive occurrence of an unknown and if another subgoal \( \vdash \Delta_2 \) contains a negative occurrence of the same unknown, then \( \vdash \Delta_1, \Delta_2 \) has size strictly less than \( \vdash \exists X F, \Lambda \).

**Proof 2.3.** We prove that \( G \) has property \( \Pr \) just above by induction on its rank \( n \) in the phase of decomposition at one go. The goal \( \{ \vdash F[\bar{X}/X], \Lambda \} \) which opens the phase has property \( \Pr \). Now, we assume that, at any step \( n \) of the phase of decomposition at one go, the current goal verifies \( \Pr \) and we want to prove that it also verifies \( \Pr \) at the next step if this step still exists in this phase. We distinguish different cases according to the MALL2 rule that is applied in step \( n+1 \).

**the \( \oplus_1, \oplus_2, \forall, \exists \)-rules.** The main formula of the active subgoal contains all its unknowns according to Lemma 2.2. Step \( n+1 \) consists in decomposing this formula into a smaller formula which guarantees that property \( \Pr \) still holds.

**the \( \otimes \)-rule.** The active subgoal \( \vdash \Delta_1, F_1 \otimes F_2, \Delta_2 \) is decomposed into two subgoals \( \vdash \Delta_1, F_1 \) and \( \vdash F_2, \Delta_2 \). By examining all cases, we show that property \( \Pr \) holds. In
particular, when $F_1$ and $F_2$ contain the same unknown with opposite polarities, the sequent $\vdash A_1, F_1, F_2, A_2$ has size less than $\vdash \exists X F, A$ because, by induction hypothesis, $\vdash A_1, F_1 \otimes F_2, A_2$ has size less than $\vdash \exists X F, A$.

**the $\exists$-rule.** The active subgoal $\vdash F_1 \exists F_2, A'$ becomes $\vdash F_1, F_2, A'$ which has the same size, if we define the size of the sequent as the length of the string that represents it. Thus, property Pr still holds.

**the $\&$-rule.** The active subgoal $\vdash F_1 \& F_2, A'$ is decomposed into two subgoals $\vdash F_1, A'$ and $\vdash F_2, A'$. Because of Condition 3, $F_1$ and $F_2$ cannot contain occurrences of the same unknown with opposite polarities. Thus, we do not have to verify that $\vdash F_1, A', F_2, A'$ has size strictly less than $\vdash \exists X F, A$ and the current goal still has property Pr.  □

Now, with the two previous lemmas, we can establish the main theorem.

**Theorem 2.1.** The procedure $\mathcal{P}_2$ is a deduction procedure which is a decision procedure.

**Proof 2.4.** We have first to prove that $\mathcal{P}_2$ is a deduction procedure, i.e. all its elements are deductions in the sense given in Definition 1.1 (soundness). Then, we have to prove termination and completeness of $\mathcal{P}_2$ in order to show that it is a decision procedure.

**Soundness.** For any deduction in $\mathcal{P}_2$, we have to show that all its steps are sound. It is obvious for the steps corresponding to case (a) of Definition 2.3. For steps corresponding to case (b), the only critical situation is when the applied rule is the $\forall$-rule. In this situation, the fresh variable $Y$ that appears in the new subgoal must not be free in the old subgoal. Now, this subgoal may contain unknowns which will be instantiated later when the principle of elimination of unknowns is applied. The formulas that instantiate such unknowns must not contain $Y$ as a free variable. Let $\overline{X}$ be any unknown that is present in the concerned subgoal. This unknown will be instantiated in a step of elimination of unknowns. At this moment, the active subgoals have the form $\vdash \overline{X}, A_1 \cdots \vdash \overline{X}, A_n \vdash \overline{X}', A'_1 \cdots \vdash \overline{X}', A'_m$. The elimination of $\overline{X}$ amounts to giving it the value $\mathcal{S}(A'_1) \& \cdots \& \mathcal{S}(A'_m)$. Now, according to Lemma 2.2, $A'_1, \ldots, A'_m$ do not contain $Y$ and thus, the formula that replaces $\overline{X}$ also. As a consequence, the eigenvariable condition linked to the apparition of $Y$ is respected.

Finally, steps produced in case (c) are sound because of the right to left direction of Lemma 2.1.

**Termination.** Proving termination for $\mathcal{P}_2$ consists first in proving that every deduction of $\mathcal{P}_2$ is finite.

In such a deduction, let us consider a phase of decomposition at one go of a sequent $\vdash \exists X F, A$. This phase is finite because it is limited by the size of $F$. Let $G$ be the last goal of this phase. All unknowns that are present in $G$ are eliminated just afterwards in a finite number of steps to produce a goal $G'$ without unknowns, unless the deduction terminates before. According to Lemma 2.3, every pair ($\vdash Y, A_1$, $\vdash
\( \overrightarrow{Y}, A_2 \) of subgoals of \( G \) is such that \( \vdash \overrightarrow{Y}, A_1, \overrightarrow{Y}, A_2 \) has size less than \( \vdash \exists X F, A \).
A fortiori, each corresponding new subgoal \( \vdash A_1, A_2 \) of \( G' \) has also size less than \( \vdash \exists X F, A \).

Now, the steps that are not inside a phase of decomposition at one go or inside a phase of elimination of unknowns, consist in replacing a subgoal with one or two new subgoals which are smaller.
Therefore, in all cases, a deduction of \( P_2 \) is a succession of phases where one subgoal is replaced with a finite number of new subgoals with a lesser size. This guarantees that such a deduction is finite.
To achieve the proof of termination for \( P_2 \), we have to prove that there is a finite number of deductions in \( P_2 \) for a given goal. This immediately follows from the fact there is only a finite number of possible deduction steps from a given goal.

**Completeness.** To prove completeness of \( P_2 \), we have to show that, for every provable sequent \( \vdash A \) of MALL', there is a successful deduction in \( P_2 \) that begins with \( \{\vdash A\} \). For this, we use induction over the size of \( \vdash A \).
Since \( P_1 \) is complete, there exists a successful deduction \( D \) in \( P_1 \) which starts from \( \{\vdash A\} \). The subgoals that result from the first step have size strictly less than \( \vdash A \).
Thus, by induction hypothesis, they are provable with deductions in \( P_2 \). If the first step of \( D \) is not an application of the \( \exists \)-rule, this is the beginning of a deduction which belongs to \( P_2 \) and we can continue it to prove \( \vdash A \) within \( P_2 \).
If the first step of \( D \) is an application of the \( \exists \)-rule, let \( \exists X F \) be the formula that is decomposed in this application. According to the occurrences of \( X \) in \( F \), four cases are possible.

\( X \) is not free in \( F \). By induction hypothesis, the subgoal that results from the first step of \( D \) is provable in a deduction of \( P_2 \). By adding the first step of \( D \) in front, we obtain a deduction of \( \vdash A \) that belongs to \( P_2 \).

\( F \) contains only positive occurrences of \( X \). By induction hypothesis, the subgoal that results from the first step of \( D \) is provable in a deduction \( D' \) of \( P_2 \). If we replace the formula \( G \) that instantiates \( X \) with \( T \) in all steps of \( D' \) that contain \( G \) and if we delete all steps that result from the decomposition of \( G \), we obtain a new deduction of \( P_2 \). By adding the first step of \( D \) in front, we obtain a deduction of \( \vdash A \) that belongs to \( P_2 \).

\( F \) contains only negative occurrences of \( X \). We proceed in a way similar to the previous one, changing \( T \) into 0.

\( F \) contains both positive and negative occurrences of \( X \). In \( D \), we replace the formula that instantiates \( X \) with \( X \) until it is decomposed. Then, we consider all applications of the \( \exists \)-rule that entails the decomposition of formulas \( \exists Y G \) in which \( X \) is present. According to the polarity of the occurrences of \( Y \), we replace the formula that instantiates \( Y \) with \( T \), 0 or \( \overrightarrow{Y} \) until it is decomposed. In the first two cases, we delete the steps in which the formula is decomposed. We iterate the process until it terminates, which is the case because it is limited by the size of \( F \). We aim to re-order \( D \) in such way that it begins with a phase of decomposition at one go until
all unknowns appear un-nested in the current goal. Since deduction $D$ belongs to $\mathcal{P}_1$, the first application of the $\exists$-rule opens a phase of chaining decomposition of the formula $\exists x F$ which is possibly followed by a phase of immediate decomposition. At the beginning, they merge into a phase of decomposition at one go. Because of Condition 2 in the definition of the MALL2' syntax, this phase can be broken into three ways.

1. An application of the $\otimes$, $\oplus_1$ or $\oplus_2$-rule for decomposing a formula that contains an unknown continues with the decomposition of a subformula that does not contain any unknown. As the resulting subgoal does not contain any unknown because of Lemma 2.2 and it is independent of the others subgoals, we can postpone the steps of $D$ that stem from its decomposition as far as the end of $D$.

2. An unknown is decomposed. As for the previous case, we can postpone the steps of $D$ that stem from its decomposition as far as the end of $D$.

3. An application of the $\exists$, $\&$ or $\forall$-rule for decomposing a formula that contains an unknown continues with the decomposition of another formula that does not contain any unknown by application of one of the same rules. This step belongs necessary to a phase of immediate decomposition. Because of Condition 2 in the definition of the MALL2' syntax, all following steps in which formulas with unknowns are decomposed are applications of the same rule in the same phase so that we can make them earlier in order to constitute a phase of decomposition at one go.

In the last case above, the result of the transformation is a deduction that starts with a phase of decomposition at one go which ends with a goal $G'$ in which all unknowns have been un-nested. In the first two cases, we iterate the transformation of the deduction until we obtain a similar deduction. Then, by iterated application of the principle of elimination of unknowns, we extend the deduction which ends with $G'$ to a goal $G''$ without unknowns. According to the left-to-right direction of Lemma 2.1, the provability of the subgoals of $G'$ entails the provability of the subgoals of $G''$. Because of Lemma 2.3, these subgoals have size strictly less than $\vdash A$. Therefore, by induction hypothesis, each one is provable in a deduction of $\mathcal{P}_2$ so that, by concatenation, we can build a deduction of $\vdash A$ which belongs to $\mathcal{P}_2$.

MALL2' contains the multiplicative and additive fragment of propositional linear logic (MALL0) which is PSPACE-complete [9]. Thus, MALL2' is PSPACE-hard and it remains to show that it is PSPACE to prove that it is PSPACE-complete.

**Theorem 2.2.** MALL2' is PSPACE-complete.

**Proof 2.5.** We have only to prove that MALL2' is PSPACE. To implement the procedure $\mathcal{P}_2$, we need a stack the top of which contains the current goal. At the beginning, the current goal is the MALL2' sequent to prove. We assume that its size is $n$. Then, according to the nature of the deduction step, four cases are possible:

1. it is an application of a MALL2 one-premiss rule; we stack the new goal; its size is strictly less than $n$ if we consider that an unknown has the size of an atomic formula;
2. it is an application of the $\otimes$-rule. In the goal, we replace the conclusion of the inference with its two premises and we push the new goal on the stack; since the partition of the context is non-deterministic, we order its possible partitions and we keep the last selected partition in memory on the top of the stack with the new goal; for this, the maximum breadth of the stack that is necessary is $2n, n$ for the goal and $n$ for the last selected partition;

3. it is an application of the $\&$-rule; in the goal, we replace the conclusion of the inference with its left premiss and we stack this modified goal; then, when this premiss will be proved, we will try the right premiss;

4. it is an elimination of an unknown; with the sequent that replaces the sequent or the two sequents where the unknown is eliminated, we obtain a new goal which is stacked;

5. when a subgoal is an axiom, we delete it from the top of the stack and if the top of the stack becomes empty, we backtrack to the last application of the $\&$-rule where the right premiss remains to prove; if such goal does not exist, the deduction succeeds;

6. when a subgoal contains only atomic formulas and is not an axiom, the deduction fails and we backtrack to the last goal for which another choice is possible.

In this implementation of the procedure $\mathcal{P}_2$, the principle of elimination of unknowns is sequentialised to occupy a minimum of space. This is performed by sequentialising the proofs of both premises in the application of the $\&$-rule. It is possible because of Condition 3 in the definition of the MALL2' syntax: if a premiss of a $\&$-inference contains an unknown $\overline{X}$, the other premiss does not contain $\overline{X}^{\perp}$. Thus, by keeping only one premiss at the same time, we do not restrict the new sequents generated by elimination of unknowns.

The maximum height of the stack is $n$ and its maximum breadth is $2n$, so that the space that is necessary to $\mathcal{P}_2$ is $2n^2$.

3. Transposition to the intuitionistic framework

This decidability result can be transposed to the multiplicative and additive fragment of second-order intuitionistic linear logic (IMALL2) via a translation of IMALL2 into MALL2 which uses polarities [8]. Each IMALL2 formula has two translations into MALL2, a positive translation and a negative translation which correspond to its position as a member of the antecedent or the succedent of a sequent. Both translations are built inductively on the structure of the formulas according to the following rules:

$$
\begin{align*}
A^+ &= A, & A^- &= A^\dag, \\
X^+ &= X, & X^- &= X^\perp, \\
(F \otimes G)^+ &= F^+ \otimes G^+, & (F \otimes G)^- &= F^- \otimes G^-, \\
(F \multimap G)^+ &= F^- \otimes G^+, & (F \multimap G)^- &= F^+ \otimes G^-, \\
\end{align*}
$$
Each sequent $F_1, \ldots, F_n \vdash G$ of IMALL2 is translated into the sequent $\vdash F_1^-, \ldots, F_n^-, G^+$ of MALL2 and it is provable in IMALL2 iff its translation is provable in MALL2. Conversely, from any sequent of MALL2, we can select a formula as the positive translation and the other formulas as the negative translations of IMALL2 formulas and we obtain a sequent of IMALL2 the translation of which into MALL2 is the initial sequent.

As a consequence, the sequents of IMALL2, for which the negative translation of formulas which are members of the antecedent and the positive translation of the unique formula which represents the succedent are in MALL2', constitute a decidable fragment of IMALL2. So, in this fragment, the syntax of formulas is double: if a formula belongs to the antecedent of a sequent, it respects a certain form and if it constitutes the succedent of a sequent, it respects another form.

**Example 3.1.** The formula $C = \forall X(X \rightarrow (X \otimes X))$ is added to the antecedent of sequents to simulate contraction in second order intuitionistic logic inside IMALL2 [10]. The negative translation of $C$ in MALL2 is the formula $\exists X(X \otimes (X_\perp \otimes X_\perp))$ which is not a member of MALL2' because it does not verify Condition 1. This observation is logical because contraction causes the undecidability of second-order intuitionistic logic.

Dropping commutativity from MALL2 does not change its decidable fragment and if we restrict ourselves to the multiplicative part, we obtain a decidable fragment of the second order Lambek Calculus (LK2). We note that the five formulas of Emms [2] belong to this fragment.

**4. Conclusion**

We have found four conditions that define a syntactic restriction of MALL2 which is decidable but we are not sure that all are necessary. We suspect that only Conditions 1 and 3 are essential. Condition 2 allows existential formulas to be decomposed at one go. If we drop it, some inferences which do not lead to emergence of unknowns could interleave with inferences that lead to the goal. These are a problem if their type is & as Example 2.2 shows but the question remains open of finding a deduction strategy that gets round this obstacle. A positive answer to this question implies that Condition 4 can also be dropped because it is linked to Condition 2.
Acknowledgements

Thanks are due to Adam Cichon who reread this article and to the anonymous referee who discovered the mistake in the proof of the main theorem.

References