A UNIFORM PROOF-THEORETIC INVESTIGATION OF LINEAR LOGIC PROGRAMMING

J.A. Harland
University of Melbourne
Australia

D.J. Pym
University of Birmingham*
England, U.K.

November 7, 1994

Abstract

In this paper we consider the problem of identifying logic programming languages for linear logic. Our analysis builds on a notion of goal-directed provability, characterized by the so-called uniform proofs, previously introduced for minimal and intuitionistic logic. A class of uniform proofs in linear logic is identified by an analysis of the permutability of inferences in the linear sequent calculus. We show that this class of proofs is complete (for logical consequence) for a certain (quite large) fragment of linear logic, which thus forms a logic programming language. We obtain a notion of resolution proof, in which only one left rule, of clause-directed resolution, is required. We also consider a translation, resembling those of Girard, of the hereditary Harrop fragment of intuitionistic logic into our framework. We show that goal-directed provability is preserved under this translation.

1 Introduction

An interesting recent development in logic of some significance for theoretical computer science is *linear logic* [7]. We discuss the natural question of the significance of linear logic for logic programming. In particular, we investigate the definition of logic programming languages within linear logic and consider the characterization of their novel features by means of an analysis of the linear sequent calculus.

The two-sided linear sequent calculus is given in Appendix A. We let ϕ, ψ, χ and ξ range over (linear) formulae and let Γ, Γ', *etc.*, range over antecedents and Δ, Δ', *etc.*, range over succedents. We work with the cut-free linear sequent calculus throughout. We consider antecedents and succedents to be multisets of formulae and so are able to suppress all occurrences of the exchange rules in proofs. If $\mathcal{F} = \{\phi_1, \ldots, \phi_m\}$ is a multiset of formulae then we write $\bigotimes_{\phi \in \mathcal{F}} \phi$ to denote the formula $\phi_1 \otimes \ldots \otimes \phi_m$. We assume that the reader is familiar with treatment of a logic program and a goal as the antecedent and succedent of a certain sequent.

We take as our point of conceptual departure the *uniform proofs* of Miller *et al.* [19], [17] and the class of hereditary Harrop formulae of intuitionistic logic. In particular, we assume that the characteristic feature of logic programming is goal-directed proof-search. More precisely, there is a search operation corresponding to each logical connective and when searching for a proof of a given goal one applies the search operation that corresponds to the outermost connective of that goal; then to the outermost connective of each subgoal so generated, *etc.*. Furthermore, we assume that it must be possible to rewrite the program so that just one left rule, an appropriate notion of *resolution* rule, is required. We stress that our notion of resolution is an *analytic* one; *cf.* classical Horn clause resolution, in which resolution amounts to cut together with unification.

In linear logic, the identification of a class of computationally appealing proofs, uniform proofs, is somewhat more intricate than in logics that have been considered previously. In first-order hereditary Harrop formulae [19], [17], the search operation corresponding to each connective that can occur in a goal is given by

*The research reported herein was carried out whilst Pym was affiliated to the University of Edinburgh, Scotland, U.K..

the right rule for that connective. Moreover, it is easy to show that the strategy of constructing proofs (from root to leaves, regarding the rules as *reduction operators*[1]) by applying a right rule wherever one is applicable is complete. This property arises from the permutability properties of the rules of the intuitionistic sequent calculus. For example, it should be clear that an intuitionistic subproof of the form

$$\dfrac{\dfrac{\Gamma, \chi, \xi \vdash \phi \qquad \Gamma, \chi, \xi \vdash \psi}{\Gamma, \chi, \xi \vdash \phi \wedge \psi}\ \wedge\text{-R}}{\Gamma, \chi \wedge \xi \vdash \phi \wedge \psi}\ \wedge\text{-L}$$

can be replaced by one of the form

$$\dfrac{\dfrac{\dfrac{\Gamma, \chi, \xi \vdash \phi}{\Gamma, \chi \wedge \xi \vdash \phi}\ \wedge\text{-L} \qquad \dfrac{\Gamma, \chi, \xi \vdash \psi}{\Gamma, \chi \wedge \xi \vdash \psi}\ \wedge\text{-L}}{\Gamma, \chi \wedge \xi \vdash \phi \wedge \psi}\ \wedge\text{-R}.$$

It turns out that for the whole intuitionistic hereditary Harrop fragment, in which just $\wedge$, $\supset$ and $\forall$ may occur negatively (the *definite formulae*) and $\wedge$, $\vee$, $\supset$, $\forall$ and $\exists$ may occur positively (the *goal formulae*), the strategy of applying right rules wherever they are applicable is complete. Furthermore, it is possible, for such classes of definite formulae and goal formulae, to encode all necessary instances of left rules in these proofs by a single, *clause-directed* left rule, giving rise to the appropriate notion of *resolution* proof. This is achieved by rewriting the antecedents of the sequents in a certain *clausal form*. For example, for a simple logic of Horn clauses, a restriction of hereditary Harrop formulae, the resolution rule can be taken to have the following form:

$$\dfrac{\Gamma \vdash G_1[\vec{t}/\vec{x}] \ldots \Gamma \vdash G_n[\vec{t}/\vec{x}]}{\Gamma \vdash A},$$

where the clause $(G_1(\vec{x}) \wedge \ldots \wedge G_m(\vec{x})) \supset B(\vec{x}) \in \Gamma$ and $B[\vec{t}/\vec{x}] = A$, for some substitution $[\vec{t}/\vec{x}]$. Note that by restricting the class of goal formulae in this setting, it is possible to eliminate right rules entirely, so that the only rule of any kind that is needed is the resolution rule.

The main novelty in linear logic is that for the desired classes of definite formulae and goal formulae, it may be necessary, under certain circumstances, to apply a left rule in the middle of a sequence of right rules in order to maintain completeness. For example, consider the sequent $\phi \otimes \psi \vdash \phi \otimes \psi$. Clearly, there is a proof in which the $\otimes$-R rule occurs above the $\otimes$-L rule, *i.e.*

$$\dfrac{\dfrac{\phi \vdash \phi \qquad \psi \vdash \psi}{\phi, \psi \vdash \phi \otimes \psi}\ \otimes\text{-R}}{\phi \otimes \psi \vdash \phi \otimes \psi}\ \otimes\text{-L}.$$

but we cannot apply the rules in the reverse order, as neither $\vdash \phi$ nor $\phi \otimes \psi \vdash \psi$ is provable. Hence, we cannot always push the $\otimes$-L above the $\otimes$-R rule; but the cases in which this is not possible can be classified.

A similar example occurs for the $C!$-L rule. Consider the sequent $!\phi \vdash \phi \otimes \phi$. Clearly, there is a proof in which the $\otimes$-R rule precedes the $C!$-L rule, *i.e.*

[1] Kleene [14] explains this in the case of classical predicate calculus.

$$\dfrac{\dfrac{\dfrac{\phi \vdash \phi}{!\phi \vdash \phi}\ !\text{-L} \qquad \dfrac{\phi \vdash \phi}{!\phi \vdash \phi}\ !\text{-L}}{!\phi, !\phi \vdash \phi \otimes \phi}\ \otimes\text{-R}}{!\phi \vdash \phi \otimes \phi}\ C!\text{-L}.$$

but we cannot apply the rules in the reverse order, as $\vdash \phi$ is not provable. Thus, as in the previous case, we cannot always push the $C!$-L above the $\otimes$-R rule; but the cases in which this is not possible can be classified.

Similar comments apply to the !-R rule, for which, due to the syntactic restrictions which are placed on the premiss of the rule, there is a larger number of exceptions. For example, consider the sequent $!\phi \otimes !\psi \vdash !\psi$, which has a proof containing an inference of the form

$$\dfrac{\dfrac{!\phi, !\psi \vdash \psi}{!\phi, !\psi \vdash !\psi}\ !\text{-R}}{!\phi \otimes !\psi \vdash !\psi}\ \otimes\text{-L}.$$

Again, we cannot apply the rules in the reverse order, as that would create a formula on the left without ! as its outermost connective.

One solution to this problem would be to exclude $\otimes$ and ! from the class of definite formulae in linear logic. However, this is both undesirable and unnecessary. It turns out that these exceptions to the strategy of applying right rules wherever they are applicable can be eliminated in the next step of our analysis, in which, just as in the setting of intuitionistic hereditary Harrop formulae, we rewrite the antecedents to a certain clausal form for which the appropriate notion of resolution rule is complete.

Thus we show how an analysis of the permutability properties of the two-sided linear sequent calculus can be used to determine fragments of linear logic for which a suitable notion of resolution proof can be defined. A summary of this analysis consists of essentially two, mutually dependent, steps: (i) The identification of the class of formulae such that the strategy of constructing proofs (considering rules to be reduction operators, in the sense of Kleene [14]) by applying right rules wherever they are applicable is complete, subject to certain exceptions to this strategy, which must be handled, *i.e.*, made goal-directed, by the second step. This class of proofs, constructed by the application of the right rules wherever possible, subject to the exceptions to be handled by step (ii), is the appropriate class of uniform proofs; (ii) The exploitation of the structural properties of the antecedents of linear sequents so that exceptions to the right rule-first strategy can be eliminated and so that we can invoke a single, clause-directed resolution rule whilst retaining (soundness and) completeness with respect to the linear sequent calculus. This is achieved by introducing a mapping which reduces antecedents to multisets of *clauses*, see Definition 3.2.

Of course, there are other, more pragmatically motivated, influences on the choices of definite formulae; indeed on the choices of goal formulae, *q.v.* [19], [17]. We remark here that if we were to restrict definite formulae to be just the clauses permitted by (ii), above, we should be forced to constrain goal formulae rather more than is essential, see Definition 2.13. It remains an open problem to determine the maximal class of linear formulae for which suitable notions of uniform proof are complete.

Aspects of this work have been presented in [10], [11], [12] and [21]. In [10] and [21] an analysis of the relevant permutability properties of the linear sequent calculus is given, together with an account of resolution proof for a classes of definite formulae not including the multiplicative disjunction, ✠, and goal formulae not including the exponential, !. These papers also include the translation of intuitionistic hereditary Harrop formulae into linear definite formulae and goal formulae, for which the restricted classes of formulae are adequate, and discuss the preservation of goal-directed provability. In [11], we present a summary of the analysis of [10]. In [12] (and [21]) our analysis is extended to include the connective, ✠, in definite formulae

and exponentials, ? and !, in goal formulae (and a few other minor extensions). These extensions introduce considerable complications to the analysis.

This paper, which expounds our analysis in some depth, is organized as follows. In § 2, we give an analysis of the relevant permutability properties of linear logic and show how these can be used to determine the appropriate notion of uniform proof. In particular, we show that this class of proofs is complete for a certain fragment of the logic. In § 3, we use this analysis to determine the appropriate notion of resolution proof, *i.e.* one in which the left rules are replaced by a single rule, and show that such proofs are (sound and) complete for the same fragment of the logic. In § 4, we show how first-order hereditary Harrop formulae can be recovered, by the use of a particular translation of that class of formulae into linear logic, thus showing that our analysis is indeed a natural generalization of previous work. In § 5, we conclude with a brief discussion of some of the topics of our current related research.

2 Uniform Linear Proofs

2.1 Permutability Properties

In this section we present those permutability properties of the linear sequent calculus that we require in order, in § 2.2, to develop a notion of uniform proof for linear logic. An analysis of the permutability properties of the classical sequent calculus is provided by Curry in [5] and for the classical and intuitionistic systems by Kleene in [15]. The permutability properties of the one-sided linear sequent calculus have been studied by Patrick Lincoln.

PROPOSITION 2.1 *If there is a proof of a given endsequent in which there is a subproof in which either* $\mathrm{R}^{\perp}$, $\perp$-R, $\with$-R, $\oplus$-R, $\multimap$-R, ?-R, W?-R, C?-R, $\bigwedge$-R, $\bigvee$-R *or* &-R *precedes either* $\perp$-L, $\mathbf{1}$-L, &-L, $\otimes$-L, $\with$-L, !-L, W!-L, C!-L, $\bigwedge$-L *or precedes* $\multimap$-L *on the right, then there is proof of the same endsequent in which this subproof is replaced by a subproof in which the left rule precedes the right rule.*

Next we consider the $\otimes$-R rule. As noted above, the permutability results are somewhat more intricate for this rule than for the other right rules. In particular, the rules $\otimes$-L and C!-L need a more careful analysis. However, for the other rules, a result similar to the one above can be obtained.

PROPOSITION 2.2 *If there is a proof of a given endsequent in which there is a subproof in which* $\otimes$-R *precedes either* $\mathbf{1}$-L, $\with$-L, &-L, !-L, W!-L, $\bigwedge$-L *or precedes* $\multimap$-L *on the right, then there is a proof of the same endsequent with this subproof replaced by a subproof in which the left rule precedes* $\otimes$-R.

As noted above, the omission of the $\otimes$-L rule and the C!-L rule from Proposition 2.2 is necessary in that there are some cases in which either the $\otimes$-R rule or the C!-L rule must precede either the $\otimes$-L rule in order for a proof to exist. However, in both cases similar results hold under certain conditions; these conditions are precisely stated below.

PROPOSITION 2.3 *If there is a proof of a given endsequent in which there is a subproof of the form*

$$\dfrac{\dfrac{\Gamma \vdash \phi, \Delta \qquad \Gamma' \vdash \psi, \Delta'}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\ \otimes\text{-R}}{\Gamma'', \chi \otimes \xi \vdash \phi \otimes \psi, \Delta, \Delta'}\ \otimes\text{-L},$$

where $\{\chi, \xi\} \subseteq \Gamma \cup \Gamma'$, *then if* $\{\chi, \xi\} \subseteq \Gamma$ *or* $\{\chi, \xi\} \subseteq \Gamma'$ *there is a proof of the same endsequent with this subproof replaced by a subproof in which the* $\otimes$-L *rule precedes the* $\otimes$-R *rule.*

PROPOSITION 2.4 *If there is a proof of a given endsequent in which there is a subproof of the form*

$$\dfrac{\dfrac{\Gamma \vdash \phi, \Delta \qquad \Gamma' \vdash \psi, \Delta'}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\ \otimes\text{-R}}{\Gamma'', !\chi \vdash \phi \otimes \psi, \Delta, \Delta'}\ C!\text{-L},$$

where $\{!\chi, !\chi\} \subseteq \Gamma \cup \Gamma'$*, then if* $\{!\chi, !\chi\} \subseteq \Gamma$ *or* $\{!\chi, !\chi\} \subseteq \Gamma'$ *there is a proof of the same endsequent with this subproof replaced by a subproof in which the* $C!$*-L rule precedes the* $\otimes$*-R rule.*

An important omission from the above analysis is the !-R rule. The main reason for this is that this rule allows far fewer permutability results than the other right rules do, because of the syntactic restrictions in the premiss of the rule. This means that the permutability properties in this case have a different flavour, being somewhat more restrictive than those for the other rules. The result below shows the (few) permutations that are possible for !-R.

PROPOSITION 2.5 *If there is a proof of a given endsequent in which there is a subproof in which* !*-R precedes either* **1**-L, !-L, *W*!-L *or C*!-L*, then there is a proof of the same endsequent with this subproof replaced by a subproof in which the left rule precedes* !-R*.*

In all other cases we cannot permute the !-R rule below a left rule, as then the syntactic restrictions of the !-R rule will not be satisfied. For similar reasons, the !-R can never precede the ?-L rule.

LEMMA 2.6 *There can be no subproof in which the* !-R *rule precedes the* ?-L *rule, nor any subproof in which one of the rules* $\bot$-L, $\Cross$-L, &-L, $\otimes$-L, $\oplus$-L, $\multimap$-L, $\bigwedge$-L, $\bigvee$-L *or* ?-L *precedes the* !-R *rule.*

Thus the !-R rule is significantly more restrictive than the other rules.

We now turn to the analysis of the rules for constants for the cases not covered above. This analysis is similar in nature to the !-R analysis above, due to the restrictions in the premisses of some of the rules. Indeed, it is these restrictions in the rules **1**-R, $\mathrm{L}^{\bot}$, $\top$-R and **0**-L which make them difficult to incorporate into the mainstream analysis, as **1**-L and $\mathrm{R}^{\bot}$ were. Again there are just certain combinations of interest to us and some cases which will never arise.

LEMMA 2.7 *There can be no subproof in which the* **1**-R *rule precedes* $\mathrm{L}^{\bot}$, $\Cross$-L, &-L, $\otimes$-L, $\oplus$-L, !-L, ?-L, *C*!-L, $\bigwedge$-L, $\bigvee$-L *or* $\multimap$-L *on the right, nor any subproof in which* $\mathrm{L}^{\bot}$ *or* **0**-L *precedes any right rule.*

Hence we need not consider the permutability properties of the two rules $\mathrm{L}^{\bot}$ and **0**-L any further. On the other hand, relevant cases do arise for the $\top$-R rule; in fact, no left rule can precede this rule, as the $\top$-R rule requires an empty premiss. However, it should be clear that an occurrence of $\top$-R followed by any one of the rules $\Cross$-L, &-L, $\otimes$-L, $\oplus$-L, !-L, C!-L, W!-L, $\bigwedge$-L, $\bigvee$-L or $\multimap$-L can be replaced by a single occurrence of $\top$-R, as the succedent of the conclusion of the left rule will contain an occurrence of $\top$.

It should also be noted that $\bigvee$-L cannot be permuted above $\bigvee$-R and that $\oplus$-L cannot be permuted above any of $\oplus$-R, $\otimes$-R, W?-R, ?-R or $\bigvee$-R.

A final point to note is that, due to the equivalence in linear logic of $F^{\bot}$ and $F \multimap \bot$, it is possible to omit the $\bot$-L and $\bot$-R rules from the rest of the analysis, provided that we replace each occurrence of a formula $F^{\bot}$ with $F \multimap \bot$ and thereafter treat $\bot$ as a distinguished atom.

2.2 Uniformity

We exploit our analysis of the permutability properties of the linear connectives to obtain our notion of uniform proof for linear logic. This notion differs from that of intuitionistic logic [19, 9] rather delicately. Here we must permit certain occurrences of left rules, other than $\multimap$-L, nearer to the endsequent than certain occurrences of right rules but still retain the ability to obtain a computationally acceptable, *goal-directed* notion of *resolution proof*, *i.e.* one in which we have a single left rule corresponding to particular combinations of left rules such as $\multimap$-L, $\bigwedge$-L, $\otimes$-L *etc.*.

Specifically, we must permit occurrences of $\otimes$-L below $\otimes$-R as follows:

$$\dfrac{\dfrac{\dfrac{\vdots}{\Gamma \vdash \phi, \Delta} \qquad \dfrac{\vdots}{\Gamma' \vdash \psi, \Delta'}}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\ \otimes\text{-R}}{\Gamma'', \chi \otimes \xi \vdash \phi \otimes \psi, \Delta, \Delta'}\ \otimes\text{-L},$$
$$\vdots$$

where $\chi \in \Gamma$, $\xi \in \Gamma'$. The necessity of this is dictated by Proposition 2.3. Similarly, we must also permit contractions on the left, $C!$-L, to appear below occurrences of $\otimes$-R, as dictated by Proposition 2.4.

The notion of resolution proof is developed in § 3; however we remark here that in recovering such a notion, which is sound and complete with respect to our notion of uniform proof in which figures of the above form are permitted, we exploit the equivalence of the antecedents $\phi_1, \phi_2, \ldots, \phi_m$ and $\phi_1 \otimes \phi_2 \otimes \ldots \otimes \phi_m$. In fact, we must exploit a notion of *clausal decomposition* for the antecedents of a certain class of sequents, but the development of this notion is delayed until it is required (§ 3). For a brief example, the sequent $\phi \otimes \psi \vdash \phi \otimes \psi$ does not have a proof in which $\otimes$-R occurs below $\otimes$-L. However, observe that this sequent does have a uniform proof, in that if we first decompose the antecedent to be $\{\phi, \psi\}$, we can recover a proof in which $\otimes$-R occurs before any further activity on the left, *i.e.*, a *goal-directed* proof; see § 3. The reader is referred to Proposition 3.16 for further discussion of these issues.

Thus we come to a key definition, that of *locally LR proofs*: our proposition will be that locally LR proofs are complete with respect to a certain class of proofs in the linear sequent calculus, namely those in which there are no occurrences of the rules $\mathbf{0}$-L, $\oplus$-L, ?-L and $\bigvee$-L (and in which we transform $F^{\perp}$ to $F \multimap \perp$). Whilst this class can be enlarged by modifying the definition of locally LR proof to permit other occurrences of left rules below right rules, it does not seem to be possible to recover an acceptable, goal-directed, notion of resolution proof which is sound and complete with respect to such classes; consequently, such classes are not of interest from the point of view of logic programming. For a pedagogic purposes it is convenient to identify also the classes of RL and LR proofs.

Definition 2.8 [RL, LR and Locally LR Proofs] We define the following shapes of linear proofs:

1. A proof is RL if there is an occurrence of a right rule whose conclusion is a premiss of a left rule;
2. A proof is LR if it is not RL;
3. A proof is *locally LR* if the only occurrences of a right rule preceding a left rule are either of the form

$$\dfrac{\dfrac{\dfrac{\Xi_1}{\Gamma \vdash \phi, \Delta}\ *\text{-R} \qquad \dfrac{\Xi_2}{\Gamma', \psi \vdash \Delta'}\ \sharp\text{-L}}{\Gamma, \Gamma', \phi \multimap \psi \vdash \Delta, \Delta'}\ \multimap\text{-L},$$

where Ξ_1 and Ξ_2 are locally LR, or of the form

$$\dfrac{\dfrac{\dfrac{\Xi_1}{\Gamma \vdash \phi, \Delta} \qquad \dfrac{\Xi_2}{\Gamma' \vdash \psi, \Delta'}\ \otimes\text{-R}}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\ \otimes\text{-L},}{\Gamma'', \chi \otimes \xi \vdash \phi \otimes \psi, \Delta, \Delta'}$$

where $\chi \in \Gamma$, $\xi \in \Gamma'$ and Ξ_1 and Ξ_2 are either locally LR or of the form

$$\dfrac{\dfrac{\dfrac{\Xi_1}{\Gamma \vdash \phi, \Delta} \qquad \dfrac{\Xi_2}{\Gamma' \vdash \psi, \Delta'}}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\ \otimes\text{-R}}{\Gamma'', !\chi \vdash \phi \otimes \psi, \Delta, \Delta'}\ C!\text{-L},$$

where $!\chi \in \Gamma$, $!\chi \in \Gamma'$ and either Ξ_1 and Ξ_2 are locally LR or of the form

$$\dfrac{\dfrac{\dfrac{\Xi}{\Gamma \vdash \Delta}\ !\text{-R}}{\Gamma' \vdash \Delta'}\ \sharp\text{-L},$$

where $\sharp$-L is one of $\otimes$-L, &-L, $\multimap$-L, $\wedge$-L and ✠-L and where Ξ is either locally LR or of the form

$$\dfrac{\dfrac{}{\vdash \mathbf{1}}\ \mathbf{1}\text{-R}}{\Gamma \vdash \mathbf{1}}\ \sharp\text{-L},$$

where $\sharp$-L is either $\mathbf{1}$-L or W!-L.

Note that the first three cases above were given in [11], in which a more restrictive analysis was given. The fourth case is needed to allow for occurrences of the rule !-R, and the last is an extra initial case needed for certain occurrences of the linear constant $\mathbf{1}$.

The next proposition, and its corollary, give the completeness of locally LR proofs with respect to those linear sequent calculus proofs in which there are no occurrences of the rules $\mathbf{0}$-L, $\oplus$-L, ?-L and $\bigvee$-L.

PROPOSITION 2.9 *If* $\Gamma \vdash \Delta$ *has an* RL *proof not containing* $\mathbf{0}$-L, $\oplus$-L, ?-L *or* $\bigvee$-L, *then* $\Gamma \vdash \Delta$ *has a locally* LR *proof.*

PROOF Consider the occurrence of a right rule whose conclusion is a premiss of a left rule which is closest to the root of the proof tree. If the right rule occurs on the left of $\multimap$-L, we are done. Otherwise, if the right rule is neither $\otimes$-R nor !-R nor $\mathbf{1}$-R, then by Proposition 2.1 there is a proof with the left rule preceding the right one.

If the right rule is $\mathbf{1}$-R, then the only possibilities for the left rule are $\bot$-L, $\mathbf{1}$-L or W!-L. In the latter two cases, the proof is locally LR. In the first case, we can replace both rules by $\mathrm{L}^{\bot}$, which is clearly a locally LR proof.

If the right rule is !-R, then consider the left rule. If it is one of !-L, W!-L, and C!-L, then by Proposition 2.5, there is a proof in which the left rule precedes the right one. Now, as there are no occurrences of $\oplus$-L, ?-L or $\bigvee$-L, and !-R cannot precede ?-L, the only remaining cases are $\otimes$-L, &-L, ✠-L, $\multimap$-L and $\bigwedge$-L, in which case the proof is locally LR.

Hence let the right rule be $\otimes$-R. If the left rule is neither $\otimes$-L nor C!-L, then, by Proposition 2.2, there is a proof with the left rule preceding the right one.

If the left rule is $\otimes$-L, the proof is of the form

$$\dfrac{\dfrac{\dfrac{\Xi_1}{\Gamma \vdash \phi, \Delta} \qquad \dfrac{\Xi_2}{\Gamma' \vdash \psi, \Delta'}}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\otimes\text{-R}}{\Gamma'', \chi \otimes \xi \vdash \phi \otimes \psi, \Delta, \Delta'}\otimes\text{-L},$$

where $\{\chi, \xi\} \subseteq \Gamma \cup \Gamma'$. Now if $\chi \in \Gamma$ and $\xi \in \Gamma'$ we are done. Clearly a similar result holds when $\chi \in \Gamma'$ and $\xi \in \Gamma$. Hence, without loss of generality, we assume $\{\chi, \xi\} \subseteq \Gamma$; so $\Gamma = \chi, \xi, \Gamma'''$ and $\Gamma'' = \Gamma', \Gamma'''$. Then there is a proof of the form

$$\dfrac{\dfrac{\dfrac{\Xi_1}{\Gamma \vdash \phi, \Delta}}{\Gamma''', \chi \otimes \xi \vdash \phi, \Delta}\otimes\text{-L} \qquad \dfrac{\Xi_2}{\Gamma' \vdash \psi, \Delta'}}{\Gamma'', \chi \otimes \xi \vdash \phi \otimes \psi, \Delta, \Delta'}\otimes\text{-R}.$$

Hence let the left rule be $C!$-L; so the proof is of the form

$$\dfrac{\dfrac{\dfrac{\Xi_1}{\Gamma \vdash \phi, \Delta} \qquad \dfrac{\Xi_2}{\Gamma' \vdash \psi, \Delta'}}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\otimes\text{-R}}{\Gamma'', !\chi \vdash \phi \otimes \psi, \Delta, \Delta'}C!\text{-L},$$

where $\{!\chi, !\chi\} \subseteq \Gamma, \Gamma'$. Now if $!\chi \in \Gamma$ and $!\chi \in \Gamma'$ we are done. Clearly a similar result holds when $!\chi \in \Gamma'$ and $!\chi \in \Gamma$. Hence, without loss of generality, we assume $\{!\chi, !\chi\} \subseteq \Gamma$; so $\Gamma = !\chi, !\chi, \Gamma'''$ and $\Gamma'' = \Gamma', \Gamma'''$. Then there is a proof of the form

$$\dfrac{\dfrac{\dfrac{\Xi_1}{\Gamma \vdash \phi, \Delta}}{\Gamma''', !\chi \vdash \phi, \Delta}C!\text{-L} \qquad \dfrac{\Xi_2}{\Gamma' \vdash \psi, \Delta'}}{\Gamma'', !\chi \vdash \phi \otimes \psi, \Delta, \Delta'}\otimes\text{-R}.$$

We can repeat this process for the next rule application (towards the root of the proof tree), and so on until the resulting subproof is locally LR. We can perform this operation for all occurrences of a right rule immediately preceding a left rule and hence obtain a locally LR proof.

COROLLARY 2.10 *If there is a proof of* $\Gamma \vdash \Delta$ *which contains no occurrences of* $\mathbf{0}$-L, $\oplus$-L, ?-L *or* $\bigvee$-L, *then there is a locally* LR *proof of* $\Gamma \vdash \Delta$.

PROOF If the proof of $\Gamma \vdash \Delta$ is LR, then it is locally LR and we are done. Otherwise, the proof is RL and, by the above proposition, $\Gamma \vdash \Delta$ has a locally LR proof.

However, locally LR proofs are not quite satisfactory for our purposes, as they may involve some inessentially complicated subproofs on the right hand side of the $\multimap$-L rule. Hence we introduce the notion of a *simple* proof, *cf.* [17].

The intuition behind simple proofs is that the right hand subproof used in the $\multimap$-L rule should be trivial, *i.e.* require no effort to prove, the only subgoals being those corresponding to the antecedent of the principal, implicational, formula. For example, the following proof is not simple:

$$\dfrac{\chi \vdash \chi \qquad \dfrac{\dfrac{\phi \vdash \phi}{\phi \,\&\, \psi \vdash \phi}\ \&\text{-L}}{\chi, \chi \multimap (\phi \,\&\, \psi) \vdash \phi}\ \multimap\text{-L}.}$$

DEFINITION 2.11 [Simple Proofs] A linear proof Ξ is said to be *simple* if every occurrence in Ξ of the $\multimap$-L rule is of the form:

$$\dfrac{\begin{array}{c}\Xi' \\ \Gamma \vdash \phi, \Delta \end{array} \qquad \psi \vdash \psi}{\Gamma, \phi \multimap \psi, \Gamma' \vdash \psi, \Delta, \Delta'}$$

Henceforth, we shall call simple locally LR proofs *uniform* proofs. We require the following proposition about simple proofs.

PROPOSITION 2.12 *If there is a proof of a given endsequent in which there is a subproof in which one of* $\bot$-L, $\parr$-L, &-L, $\otimes$-L, !-L, W?-L, C?-L, $\bigwedge$-L, *precedes* $\multimap$-L *on the right such that the formulae acted on by the rules differ, then there is a proof of the same endsequent with this subproof replaced by a subproof in which the* $\multimap$-L *rule precedes the other left rule.*

Recall that our aim here is to show how it is possible to obtain goal-directed proof-search procedures for certain classes of linear formulae. To this end, we identify, below, classes of *definite formulae* D and *goal formulae* G for which uniform proofs are complete for consequences of the form $D_1, \ldots, D_m \vdash G_1, \ldots, G_n$.

Uniform proofs are intended to isolate a class of definite formulae which can be rewritten to a certain clausal form, for which resolution proof will be complete. However, even in the light of the permutability analysis, the choice of the class of definite formulae remains delicate. Two issues are noteworthy.

Firstly, it seems to be necessary to exclude $\oplus$, $\bigvee$ and ? from the the class of definite formulae. Although some useful permutability properties hold for these constructors, it does not seem to be possible, without significantly complicating the present analysis, to rewrite formulae containing them into computationally desirable clauses (*q.v.* Definition 3.1). A systematic elicitation of these points is the subject of current research; it is related to the question of what would constitute a maximal class of formulae for which goal-directed, clause-directed proof is complete (*q.v.* § 5).

Secondly, the proviso of Proposition 2.12 places an important restriction on definite formulae of the form $F_1 \multimap F_2$. For example, consider the following proof, for which the proviso does not hold:

$$\dfrac{r \vdash r \qquad \dfrac{p \vdash p \qquad q \vdash q}{p \parr q \vdash p, q}\ \parr\text{-L}}{r,\ r \multimap (p \parr q) \vdash p, q}\ \multimap\text{-L}.$$

There is no way to push the occurrence of $\multimap$-L any earlier in the proof, as then the required formula cannot be constructed. Thus in order to obtain the completeness of simple proofs, it seems necessary to require that all implications $F_1 \multimap F_2$ which occur in antecedents must be such that there are no logical connectives in F_2, so that we are left with just such formulae of the form $G \multimap A$ and $G \multimap \bot$. Note that it is also possible to permit definite formulae of the form $G \multimap \mathbf{1}$; the addition of such formulae would add little to the present analysis.

It is possible (see, for example, [13], [1]) to include formulae such as $G \multimap \bigwedge \vec{x}.A$, $G \multimap A_1 \& \ldots \& A_n$ and even $G \multimap A_1 \between \ldots \between A_n$, but such additions require a more complicated notion of simple proof and do not elucidate the present analysis. Their consideration is deferred along with the question of maximality mentioned above.

Hence we arrive at the classes of formulae defined below.

DEFINITION 2.13 [Definite Formulae and Goal Formulae] Let A range over atomic formulae. We define the classes of *definite formulae* and *goal formulae* as follows:

$$\begin{array}{lcll} \textit{Definite formulae} & D & ::= & A \mid \mathbf{1} \mid \bot \mid D \,\&\, D \mid D \otimes D \mid G \multimap A \mid G \multimap \bot \mid \bigwedge x.D \\ & & & \mid\, !\, D \mid D \between D \\ \textit{Goal formulae} & G & ::= & A \mid \mathbf{1} \mid \bot \mid \top \mid G \otimes G \mid G \oplus G \mid G \between G \mid G \,\&\, G \\ & & & \mid D \multimap G \mid \bigwedge x.G \mid \bigvee x.G \mid\, !\, G \mid ?\, G. \end{array}$$

Programs are linear antecedents that consist of just closed definite formulae and *goals* are linear succedents that consist of just closed goal formulae. We assume that all quantified variables are *standardized apart*.

Note that due to the presence of the linear constant $\bot$, these classes of formulae are of the same strength as those with the extra cases $G^\bot$ as a definite formula and $D^\bot$ as a goal formula. The resemblance of these definitions to those of hereditary Harrop formulae of intuitionistic logic should be clear.

Note also that a sequent $\mathcal{P} \vdash \mathcal{G}$, where $\mathcal{P}$ is a program (a multiset of closed definite formulae) and $\mathcal{G}$ is a goal, might also have a proof which is not simple. The following proof is such an example:

$$\dfrac{\phi \vdash \phi \qquad \dfrac{\phi \vdash \phi \qquad \psi \vdash \psi}{\phi, \phi \multimap \psi \vdash \psi}\ \multimap\text{-L}}{\phi, \phi \multimap \phi, \phi \multimap \psi \vdash \psi}\ \multimap\text{-L}.$$

However, it is clear that the sequent $\phi, \phi \multimap \phi, \phi \multimap \psi \vdash \psi$ does have a simple proof. This property holds for any sequent in which the antecedent is a program and the succedent is a goal.

PROPOSITION 2.14 *Let Γ be a multiset of definite formulae and let Δ be a multiset of goal formulae. If $\Gamma \vdash \Delta$ has a locally LR proof, then $\Gamma \vdash \Delta$ has a uniform (simple locally LR) proof.*

PROOF We proceed by induction on the number of occurrences of the $\multimap$-L rule in the proof. The base case corresponds to the occurrence(s) of $\multimap$-L which appear(s) closest to the leaves of the proof tree.

Consider the right hand sequent in the occurrence of $\multimap$-L. If this is initial, then we are done. Otherwise, consider the rule immediately preceding it. If this is a right rule, then by Propositions 2.1 and 2.2 we know that we can replace this subproof with an equivalent one in which the $\multimap$-L rule precedes the right rule. Otherwise, the preceding rule must be either $\between$-L, $\otimes$-L, $\&$-L, !-L, W!-L, C!-L, or $\bigwedge$-L and, by Proposition 2.12, we know that we can similarly replace this subproof with an equivalent one in which the $\multimap$-L rule precedes the (other) left rule. Clearly, we can repeat this process for each rule preceding the $\multimap$-L rule until there is no preceding rule, *i.e.* that the occurrence is simple. Note that the sequence of the other rules is not changed, so that the resulting proof is still locally LR.

Hence we assume that given an occurrence of the $\multimap$-L rule, all previous occurrences (*i.e.* those towards the leaves of the tree) are simple. Now, as above, if the occurrence of the $\multimap$-L rule is simple, we are done.

Otherwise, consider the previous rule on the right of the $\multimap$-L rule. As above, if either it is a right rule or any of $\Yinyang$-L, $\otimes$-L, & -L, !-L, W!-L, C!-L and $\bigwedge$-L, then there is a subproof with the order of the rules reversed. We can then proceed as above until we either arrive at an initial sequent or we find that the rule preceding $\multimap$-L on the rule is another occurrence of $\multimap$-L. By the hypothesis, we know that this must be a simple occurrence and that the proof is of the form

$$\dfrac{\Gamma' \vdash \psi, \Delta' \qquad \dfrac{\Gamma, \chi \vdash \phi, \Delta \qquad A \vdash A}{\Gamma, \chi, \phi \multimap A \vdash \Delta, A}\ \multimap\text{-L}}{\Gamma', \Gamma, \phi \multimap A, \psi \multimap \chi \vdash \Delta, \Delta', A}\ \multimap\text{-L},$$

where A is an atom or $\bot$, in which case clearly we can replace this proof with a proof of the form

$$\dfrac{\dfrac{\Gamma' \vdash \psi, \Delta' \qquad \Gamma, \chi \vdash \phi, \Delta}{\Gamma', \Gamma, \psi \multimap \chi \vdash \psi, \Delta, \Delta'}\ \multimap\text{-L} \qquad A \vdash A}{\Gamma', \Gamma, \phi \multimap A, \psi \multimap \chi \vdash \Delta, \Delta', A}\ \multimap\text{-L},$$

in which the latter application is simple; by the hypothesis the former application can also be made simple. Note that this reordering of the proof preserves the property that the proof is locally LR.

LEMMA 2.15 *Let Γ be a multiset of definite formulae and let Δ be a multiset of goal formulae. A proof of $\Gamma \vdash \Delta$ contains no occurrences of* **0**-L, $\oplus$-L, ?-L *or* $\bigvee$-L.

PROOF A simple induction on the structure of proofs.
Thus we arrive at the following theorem:

THEOREM 2.16 *Let Γ be a multiset of definite formulae and let Δ be a multiset of goal formulae. Then $\Gamma \vdash \Delta$ has a proof if and only if $\Gamma \vdash \Delta$ has a uniform (simple locally LR) proof.*[2]

PROOF By Lemma 2.15 we have that any linear proof of $\Gamma \vdash \Delta$ does not contain **0**-L, $\oplus$-L, ?-L or $\bigvee$-L; so by Corollary 2.10, $\Gamma \vdash \Delta$ has a locally LR proof. Hence by Proposition 2.14, $\Gamma \vdash \Delta$ has a uniform (simple locally LR) proof.

3 Resolution

We saw in § 2 how uniform (simple locally LR) proofs are complete for sequents $\mathcal{P} \vdash \mathcal{G}$, where $\mathcal{P}$ is a multiset of definite formulae and $\mathcal{G}$ is a multiset of goal formulae. However, uniform proofs do not provide (complete) goal-directed, clause-directed proof. There are two, mutually dependent, reasons for this. Firstly, uniform proofs permit occurrences of left rules where, for goal-directedness, we should like to apply a right rule. Secondly, uniform proofs permit the use of several left rules.

In this section we show how to rewrite the class of definite formulae to a form which, together with a single left rule of *resolution*, permits goal-directed, clause-directed proof construction. The basic idea is that clauses are structured in such a way, and the resolution rule organized in such a way, that all of the left rules required by uniform proofs are built into clause-directed resolution. At the simplest level, recall that uniform proofs permit occurrences of $\otimes$-L below $\otimes$-R,

[2] This result amounts to a partial *Permutation Theorem* for the given fragment of linear logic. The reader who is interested in Permutation Theorems in general is referred to [5] for classical logic, [22] for a variety of non-standard first-order connectives, and [20] for a (constructive) dependent type theory.

$$\dfrac{\dfrac{\phi \vdash \phi \qquad \psi \vdash \psi}{\phi, \psi \vdash \phi \otimes \psi}\ \otimes\text{-R}}{\phi \otimes \psi \vdash \phi \otimes \psi}\ \otimes\text{-L}.$$

However, recall that we can rewrite the antecedent from being the singleton multiset $\{\phi \otimes \psi\}$ to be the multiset $\{\phi, \psi\}$, so that we never consider formulae in the antecedent whose outermost connective is $\otimes$ and thereby can construct a proof in a goal-directed manner. Thus we can overcome the lack of permutability of $\otimes$-R below $\otimes$-L by exploiting the properties of the $\otimes$-L rule. The lack of permutability of $\otimes$-R below C!-L can be similarly handled.

In this manner, we can rewrite definite formulae into a form more conducive to resolution-style proof. We introduce below the class of *clauses*, a restricted form of definite formulae, as well as a mapping from definite formulae to the corresponding multiset of clauses.

As we shall see, not all of the connectives that occur in definite formulae can be eliminated in a manner as simple as that which is possible for $\otimes$. However, it is possible to exploit the structural properties of clauses and organize the resolution rule in such a way that it encodes in a single rule all of the instances of the $\otimes$-L, &-L, $\multimap$-L, !-L, C!-L and $\bigwedge$-L rules that are necessary to construct a proof.

DEFINITION 3.1 [Clause] Let A range over atomic formulae. We define the classes of *basic clauses*, *clauses* and *sub-clauses* as follows:

$$\begin{array}{lcl} \textit{Basic clauses} & B & ::= \quad \mathbf{1} \mid \bot \mid A \mid G \multimap A \mid G \multimap \bot \\ \textit{Subclauses} & S & ::= \quad B \mid S \otimes S \mid S \between S \mid S \,\&\, S \mid\, !\, S \\ \textit{Clauses} & C & ::= \quad B \mid S \between S \mid S \,\&\, S \mid\, !\, S. \end{array}$$

Note that when G is a conjunction of atoms, a basic clause in the above sense (disregarding the linear constants) is just a *Horn* clause, *q.v.* the resolution rule mentioned in § 1.

Below we show how to generate a multiset of clauses from a multiset of definite formulae via the mapping $[-]$.

DEFINITION 3.2 [Clausal Decomposition] Let $\cup$ denote multiset union. We define a mapping $[-]$ from multisets of definite formulae to multisets of clauses as follows:

$$\begin{array}{rcl} [\mathcal{D}] & =_{\text{def}} & \bigcup_{D \in \mathcal{D}} [D] \\ [\mathbf{1}] & =_{\text{def}} & \{\, \mathbf{1} \,\} \\ [\bot] & =_{\text{def}} & \{\, \bot \,\} \\ [A] & =_{\text{def}} & \{\, A \,\} \\ [G \multimap \bot] & =_{\text{def}} & \{\, G \multimap \bot \,\} \\ [G \multimap A] & =_{\text{def}} & \{\, G \multimap A \,\} \end{array}$$

$$\begin{array}{rcl} [D_1 \,\&\, D_2] & =_{\text{def}} & \{(\bigotimes_{C \in [D_1]} C) \,\&\, (\bigotimes_{C \in [D_2]} C)\} \\ [D_1 \otimes D_2] & =_{\text{def}} & [D_1] \cup [D_2] \\ [\bigwedge x.D] & =_{\text{def}} & [D] \\ [!\, D] & =_{\text{def}} & \{\, !\bigotimes_{C \in [D]} C \,\} \\ [D_1 \between D_2] & =_{\text{def}} & \{(\bigotimes_{C \in [D_1]} C) \between (\bigotimes_{C \in [D_2]} C)\}, \end{array}$$

where variables x that occur in $\mathcal{D}$ and *outwith* the scope of any ! are marked as *global* variables and where variables x that occur in $\mathcal{D}$ *within* the scope of some ! are marked as *local*.

Note that basic clauses are not affected by this mapping, *i.e.* that $[B] = \{B\}$ when B is a basic clause. Thus the decomposition $[\mathcal{P}]$ of a program $\mathcal{P}$ is a multiset of clauses, but not necessarily basic clauses. For example, if the program $\mathcal{P}$ is given by the single definite formula $!(\bigwedge x.(p(x) \otimes (p(x) \multimap q(x)))) \otimes r$ then we have that $[\mathcal{P}]$ is the multiset $\{!(p(x) \otimes (p(x) \multimap q(x))), r\}$.[3] It should be clear that a proof of $[\mathcal{P}] \vdash \mathcal{G}$ can be

[3] Where it is convenient, we omit the brackets { and } around multisets of formulae.

easily transformed into a proof of $\mathcal{P} \vdash \mathcal{G}$ by the insertion of the appropriate left rules and the application of appropriate substitutions.

Note that there are "free" variables in the clauses in $[\mathcal{P}]$; these are variables that are quantified in $\mathcal{P}$. Now we can think of the scope of these free variables as being "global", in that all occurrences of the variable must be updated consistently. Due to the possibility of splitting the program during execution, this may mean updating variables simultaneously across several branches of the proof.[4] This

leads us to the definitions that follow.

DEFINITION 3.3 [Global and Local Variables, Distinctness] Let $\mathcal{P}$ be a program. If x is a free variable in $[\mathcal{P}]$, then it is a *global variable* if it occurs outside the scope of any ! in P. Otherwise, x is a *local variable*.

Let $\mathcal{D}$ be a multiset of definite formulae. We say $\mathcal{D}'$ and $\mathcal{D}''$ are *distinct copies of* $\mathcal{D}$ if $\mathcal{D}'$ and $\mathcal{D}''$ are obtained from $\mathcal{D}$ by renaming the free variables of $\mathcal{D}$ so that $\mathcal{D}'$ and $\mathcal{D}''$ have no free variable names in common.

In fact, these properties are enforced by our technique of standardizing apart all bound variables, but this (rather minor) point is clarified by this definition.

We shall find it convenient to divide an antecedent into two parts — those formulae which are of the form $!F$ and those which are not.

DEFINITION 3.4 [Antecedent Division] Let $\cup$ denote multiset union and let $\mathcal{D}$ be a multiset of definite formulae. We define two multisets $\mathcal{D}^E$ and $\mathcal{D}^L$ such that $\mathcal{D} = \mathcal{D}^E \cup \mathcal{D}^L$ as follows:

$$\begin{array}{lll} \mathcal{D}^E & =_{\text{def}} & \bigcup_{D \in \mathcal{D}} \{\, D \mid \textit{the outermost connective of } D \textit{ is } ! \textit{ or } D = \mathbf{1} \,\} \\ \mathcal{D}^L & =_{\text{def}} & \bigcup_{D \in \mathcal{D}} \{\, D \mid \textit{the outermost connective of } D \textit{ is not } ! \textit{and } D \textit{ is not } \mathbf{1} \}. \end{array}$$

Similarly, it is convenient to divide a succedent into formulae which are of the form $?F$ and those which are not.

DEFINITION 3.5 [Succedent Division] Let $\cup$ denote multiset union and let $\mathcal{G}$ be a multiset of goal formulae. We define two multisets $\mathcal{G}^E$ and $\mathcal{G}^L$ such that $\mathcal{G} = \mathcal{G}^E \cup \mathcal{G}^L$ as follows:

$$\begin{array}{lll} \mathcal{G}^E & =_{\text{def}} & \bigcup_{G \in \mathcal{G}} \{G \mid \textit{the outermost connective of } G \textit{ is } ? \textit{ or } G = \bot\} \\ \mathcal{G}^L & =_{\text{def}} & \bigcup_{G \in \mathcal{G}} \{G \mid \textit{the outermost connective of } G \textit{ is not } ? \textit{ and } G \textit{ is not } \bot\}. \end{array}$$

Clearly, we can think of this distinction as specifying which formulae in a program are known to be reuseable (those commencing with !) and those which are not; similarly for ? in goals. Note that we include $\mathbf{1}$ in $\mathcal{D}^E$ and $\bot$ in $\mathcal{G}^E$. This is justified by the form of the rules $\mathbf{1}$-L and $\mathrm{R}^{\bot}$, which are similar to W!-L and W?-R respectively. Note also the linear equivalences $\mathbf{1} \multimap\!\!\circ\, !\top$ and $\bot \circ\!\!\multimap ?\mathbf{0}$.

Another concept that we will find useful is that of an expansion.

DEFINITION 3.6 [Multiset Expansion] Let $\mathcal{C}$ be a multiset and let $\cup$ denote multiset union. We define $\mathcal{C}^n$ for integers $n \geq 0$ inductively as follows:

$$\mathcal{C}^0 = \emptyset \qquad\qquad \mathcal{C}^{n+1} = \mathcal{C} \cup \mathcal{C}^n.$$

DEFINITION 3.7 [Expansion] Let $\mathcal{D}$ be a multiset of clauses, let $\cup$ denote multiset union and let $!C \in \mathcal{D}$. A *$!C$-expansion* of $\mathcal{D}$ is a multiset $\mathcal{D}'$ of clauses such that $\mathcal{D}' = \mathcal{D}^L$ or $\mathcal{D}' = \mathcal{D}^L \cup \{[C]^L\}^n \cup \mathcal{E}$ for some $n \geq 1$, where $\mathcal{E}$ is an expansion of $[C]^E$.

The notion of an expansion is intended to model occurrences of the C!-L rule in the linear sequent calculus. As the W!-L and C!-L rules can only be applied to formulae of the form $!F$, when such a formula occurs in an antecedent, it can be used an arbitrary number (including 0) of times in a proof, whereas in general a formula must be used exactly once in a proof. Thus we can, intuitively, think of the formula $!D$ as a shorthand for

[4] In terms of the resolution proofs defined below, those branches that are above the same premiss of the last &-rule (Rule 7 in the definition of resolution proof, below) as the sequent to which the resolution rule is applied. The form of the &-rule in such proofs enforces the restriction of substitutions to their own &-branches.

$\{D\}^n$ where n can be any non-negative integer. The presence of such formulae in an antecedent clearly has an effect on multiplicative rules such as the $\otimes$-R rule, as when constructing a resolution proof, we must know how to "split up" the current sequent in order to find premisses of the rule which are provable. This makes it imperative to know which formulae are preceded with a !. As the definition of clauses makes clear, such formulae can occur arbitrarily deep inside a clause. Hence given a clause $!C$ we decompose it via the mapping $[-]$ to obtain $[C]$, make an arbitrary number of copies of $[C]^L$, and proceed similarly on $[C]^E$, *i.e.* the subformulae of C which commence with !.

This procedure can be simplified when more restrictive classes of formulae are used. For example, in the language of [11], clauses need only be either basic clauses or of the form $!(C_1 \otimes \ldots \otimes C_n)$, where each C_i is a basic clause. This means that ! either occurs outermost in a clause or not at all; hence it is only necessary to inspect the outermost structure of the clause to determine expansions.

Note that when making a copy in this way of a clause containing local variables, the discipline of standardizing variables apart ensures that the only variables common to the original clause and the copy are global variables.

Now when it comes to determining whether a given atom matches a clause, we need to be able to "dismantle" the clause into basic clauses. Again, this is due to the fact that a clause can be of an arbitrary depth, so that it is not obvious from the outermost structure of the clause which atoms will match it. As discussed above, the notion of expansion given above will allow us to decompose clauses whose outermost connective is !, with the two remaining cases being clauses whose outermost connective is $⅋$ or & . In order to produce the appropriate notion of resolution proof, we need to be able to encode the uses of the $⅋$-L and &-L rules into a single one.

In the former case, an inspection of the $⅋$-L rule shows that in order to decompose the clause $C_1 ⅋ C_2$ we will need to split up the program and goal into two parts, with C_1 going to one part and C_2 going to the other. As this involves more than just the program, it is clear that we cannot do this with just the mapping $[-]$, so we introduce below the notion of a *component*; the intuition for this is that a given sequent is provable precisely when its components are all provable. The precise way in which components are derived from the original sequent is, of course, governed by the $⅋$-L rule.

In the latter case, a similar inspection of the &-L rule shows that given a clause $C_1 \,\&\, C_2$ we need to be able to choose one of the two subclauses C_1 and C_2. Whilst this could be done statically, *i.e.* in the definition of the mapping $[-]$ (which in fact is what is done in [11], [12], [10]), it seems simpler to include it in the definition of a component. For similar reasons we also include an expansion of a sequent amongst its components.

Thus we arrive at the definition below. Note that we will use $\vdash_R$ for sequents in resolution proofs. We will also sometimes use $\mathcal{P} \vdash_R \mathcal{G}$ to denote that the sequent has a resolution proof; the precise meaning will always be clear from the context.

DEFINITION 3.8 [Component] Let $\mathcal{D}$ be a multiset of clauses and $\mathcal{G}$ be a multiset of goal formulae. A *multiset of C-components* of the sequent $\mathcal{D}, C \vdash_R \mathcal{G}$ is a non-empty multiset of sequents satisfying:

1. $\mathcal{D} \vdash_R \mathcal{G}$ is a C-component of the sequent $\mathcal{D} \vdash_R \mathcal{G}$ for any $C \in \mathcal{D}$;

2. $\mathcal{C}$ is a multiset of $!S$-components of the sequent $\mathcal{D}, !S \vdash_R \mathcal{G}$, where S is *not* of the form $!(S_1 ⅋ S_2)$ or $!(S_1 \,\&\, S_2)$, if $\mathcal{C}$ is a multiset of !S-components of the sequent $\mathcal{D}, !S, \mathcal{D}' \vdash_R \mathcal{G}$, where $\mathcal{D}'$ is a $!S$-expansion of $\mathcal{D}$;[5]

3. Let C be a formula either of the form $S_1 \,\&\, S_2$ or $!(S_1 \,\&\, S_2)$. Then $\mathcal{C}$ is a multiset of C-components of $\mathcal{D}, C \vdash_R \mathcal{G}$ if either $\mathcal{C}$ is a multiset of $[S_1]$-components of the sequent $(\mathcal{D} \cup \{C\})^E, [S_1], \mathcal{D}^L \vdash_R \mathcal{G}$ or $\mathcal{C}$ is a multiset of $[S_2]$-components of the sequent $(\mathcal{D} \cup \{C\})^E, [S_2], \mathcal{D}^L \vdash_R \mathcal{G}$;

4. Let C be a formula either of the form $S_1 ⅋ S_2$ or $!(S_1 ⅋ S_2)$. Then $\mathcal{C}_1 \cup \mathcal{C}_2$ is a multiset of C-components of $\mathcal{D}, C \vdash_R \mathcal{G}$ if $\mathcal{C}_1$ is a multiset of $[S_1]$-components of the sequent $(\mathcal{D} \cup \{C\})^E, [S_1], \mathcal{D}_1 \vdash_R \mathcal{G}_1, \mathcal{G}^E$ and $\mathcal{C}_2$ is a multiset of $[S_2]$-components of the sequent $(\mathcal{D} \cup \{C\})^E, [S_2], \mathcal{D}_2 \vdash_R \mathcal{G}_2, \mathcal{G}^E$, where $\mathcal{D}^L = (\mathcal{D}_1 \cup [S_1]^L) \cup (\mathcal{D}_2 \cup [S_2]^L)$ and $\mathcal{G}^L = \mathcal{G}_1 \cup \mathcal{G}_2$.

[5] Note that Case 2 of the definition of components given in [12] is mis-stated: it is a garbled version of the correct form, which amounts to that given here in the absence of subclauses of the form $!(S_1 \,\&\, S_2)$.

Where it is notationally convenient, we will write just *component* for C-component.

Note that components can be thought of as the appropriate generalization of the notion of expansion. Note also that the components of the sequents $\mathcal{P}, !D \vdash_R \mathcal{G}$ and of $\mathcal{P}, !D, !D \vdash_R \mathcal{G}$ coincide and that any component of $\mathcal{P} \vdash_R \mathcal{G}$ is also a component of $\mathcal{P}, !D \vdash_R \mathcal{G}$. Similar remarks apply for a goal of the form $?G$.

Now when it comes to the definition of the resolution rule itself, we need not only to consider the components of the sequent, but also to match a given atom with the head of some (basic) clause in one of the components. For this reason we introduce below the notion of a *resolvant*.

DEFINITION 3.9 [Initial Sequent] Let $\mathcal{D}$ be a multiset of definite formulae, $\mathcal{G}$ be a multiset of goal formulae and A and A' be an atom or a constant. We say the sequent $\mathcal{D} \vdash_R \mathcal{G}$ is *initial* if at least one of the following conditions holds:

- $\mathcal{D}^L = \{A'\}$, $\mathcal{G}^L = \{A\}$ and $A = A'[\vec{t}/\vec{x}]$;
- $\mathcal{D}^L = \emptyset$, $\mathcal{G}^L = \{A\}$ and there exists $!A' \in \mathcal{D}^E$ such that $A = A'[\vec{t}/\vec{x}]$;
- $\mathcal{D}^L = \emptyset$ and $\mathcal{G}^L = \{\, \mathbf{1} \,\}$;
- $\mathcal{D}^L = \{\bot\}$ and $\mathcal{G}^L = \emptyset$;
- $\top \in \mathcal{G}$;

DEFINITION 3.10 [Resolvant] Let $\mathcal{D}$ range over multisets of definite formulae and $\mathcal{G}$ range over multisets of goal formulae. A *resolvant*, $\mathcal{R}$, of the sequent $\mathcal{D} \vdash_R \mathcal{G}$ is a non-empty multiset of sequents satisfying at least one of the following:

1. $\mathcal{R}$ is a multiset $\{\mathcal{D}_0 \vdash_R A, G_0,\ \ \mathcal{D}_1 \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n \vdash_R \mathcal{G}_n\}$ of components of $\mathcal{D} \vdash_R \mathcal{G}$ such that $\mathcal{D}_0 \vdash_R A, \mathcal{G}_0$ is initial;
2. $\mathcal{R}$ is the multiset $\{\mathcal{D}_0[\vec{t}/\vec{x}] \vdash_R G'[\vec{t}/\vec{x}], \mathcal{G}_0\ ,\ \mathcal{D}_1[\vec{t}/\vec{x}] \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n[\vec{t}/\vec{x}] \vdash_R \mathcal{G}_n\}$, where $\{\mathcal{D}_0, G' \multimap A' \vdash_R A, \mathcal{G}_0\ ,\ \mathcal{D}_1 \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n \vdash_R \mathcal{G}_n\}$ is a multiset of components of $\mathcal{D} \vdash_R \mathcal{G}$ and $A = A'[\vec{t}/\vec{x}]$;
3. $\mathcal{R}$ is the multiset $\{\mathcal{D}_0 \vdash_R G', \mathcal{G}_0\ ,\ \mathcal{D}_1 \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n \vdash_R \mathcal{G}_n\}$, where $\{\mathcal{D}_0, G' \multimap \bot \vdash_R \bot, \mathcal{G}_0,\ \mathcal{D}_1 \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n \vdash_R \mathcal{G}_n\}$ is a multiset of components of $\mathcal{D} \vdash_R \mathcal{G}$.

Thus we come to the definition of resolution proof.

DEFINITION 3.11 [Resolution Proof] Let $\mathcal{D}$ range over multisets of definite formulae and let $\cup$ denote multiset union. A *resolution proof* is a tree, constructed from root to leaves, regulated by the following rules, subject to the side-condition that any variable x that occurs on two or more branches of the tree as a consequence of the either the resolution rule (2), the $\otimes$-rule (4) or the !-rule (12) must receive the same substitution on each branch:

1. The axiom judgement is given by:
$$\frac{}{\mathcal{D} \vdash_R \mathcal{G}}\ ,$$
where $\mathcal{D} \vdash_R \mathcal{G}$ is initial;
2. The resolution rule:
$$\frac{\mathcal{D}_1 \vdash_R \mathcal{G}_1\ \ \ldots\ \mathcal{D}_n \vdash_R \mathcal{G}_n}{\mathcal{D} \vdash_R \mathcal{G}},$$
where $\bigcup_{i=1}^{n} \{\mathcal{D}_i \vdash_R \mathcal{G}_i\}$ is a resolvant of $\mathcal{D} \vdash_R \mathcal{G}$;
3. The $R^{\bot}$-rule:
$$\frac{\mathcal{D} \vdash_R \mathcal{G}}{\mathcal{D} \vdash_R \bot, \mathcal{G}};$$

4. The $\otimes$-rule:
$$\frac{\mathcal{D}_0 \vdash_R G_1, \mathcal{G}_0 \quad \mathcal{D}_0' \vdash_R G_2, \mathcal{G}_0' \quad \mathcal{D}_1 \vdash_R \mathcal{G}_1 \ \ldots \ \mathcal{D}_n \vdash_R \mathcal{G}_n}{\mathcal{D}, C \vdash_R \ G_1 \otimes G_2, \mathcal{G}},$$
where $\{\mathcal{D}_0, \mathcal{D}_0' \vdash_R \ G_1 \otimes G_2, \mathcal{G}_0, \mathcal{G}_0'\} \ \cup \ \bigcup_{i=1}^{n} \{\mathcal{D}_i \vdash_R \ \mathcal{G}_i\}$ is a multiset of C-components of $\mathcal{D}, C \vdash_R G_1 \otimes G_2, \mathcal{G}$;

5. The $\oplus$-rule:
$$\frac{\mathcal{D} \vdash_R G_1, \mathcal{G}}{\mathcal{D} \vdash_R G_1 \oplus G_2, \mathcal{G}} \quad \frac{\mathcal{D} \vdash_R G_2, \mathcal{G}}{\mathcal{D} \vdash_R G_1 \oplus G_2, \mathcal{G}};$$

6. The $\parr$-rule:
$$\frac{\mathcal{D} \vdash_R G_1, G_2, \mathcal{G}}{\mathcal{D} \vdash_R G_1 \parr G_2, \mathcal{G}};$$

7. The &-rule:
$$\frac{\mathcal{D}' \vdash_R G_1', \mathcal{G}' \qquad \mathcal{D}'' \vdash_R G_2'', \mathcal{G}''}{\mathcal{D} \vdash_R G_1 \ \& \ G_2, \mathcal{G}},$$
where $\mathcal{D}'$ and $\mathcal{D}''$ are distinct copies of D and where $G_1', \mathcal{G}'$ and $G_2'', \mathcal{G}''$ are obtained from $G_1, \mathcal{G}$ and $G_2, \mathcal{G}$ by the applying the renamings of $\mathcal{D}'$ and $\mathcal{D}''$ respectively. Note that we maintain "global" and "local" markings;

8. The $\multimap$-rule:
$$\frac{\mathcal{D}, [D] \vdash_R G, \mathcal{G}}{\mathcal{D} \vdash_R D \multimap G, \mathcal{G}};$$

9. The $\bigwedge$-rule:
$$\frac{\mathcal{D} \vdash_R G[y/x], \mathcal{G}}{\mathcal{D} \vdash_R \bigwedge x.G, \mathcal{G}},$$
where y is not free in $\mathcal{D}$ or $\mathcal{G}$;

10. The $\bigvee$-rule:
$$\frac{\mathcal{D} \vdash_R G[t/x], \mathcal{G}}{\mathcal{D} \vdash_R \bigvee x.G, \mathcal{G}};$$

11. The ?-rule:
$$\frac{\mathcal{D} \vdash_R \{G\}^n, ?\, G, \mathcal{G}}{\mathcal{D} \vdash_R ?\, G, \mathcal{G}},$$
for some $n \geq 1$;

12. The !-rule:
$$\frac{\mathcal{D}_0 \vdash_R G, \mathcal{G}_0 \quad \mathcal{D}_1 \vdash_R \mathcal{G}_1 \ \ldots \ \mathcal{D}_n \vdash_R \mathcal{G}_n}{\mathcal{D}, C \vdash_R \ !\, G, \mathcal{G}},$$
where $\{\mathcal{D}_0 \vdash_R G, \mathcal{G}_0\} \ \cup \ \bigcup_{i=1}^{n} \{\mathcal{D}_i \vdash_R \mathcal{G}_i\}$ is a multiset of C-components of $\mathcal{D} \vdash_R \ !\, G, \mathcal{G}$ and $\mathcal{D}_0^L = \emptyset = \mathcal{G}_0^L$.

Rules 3-12 of Definition 3.11 correspond to the right rules of uniform proofs and Rule 2 corresponds to the left rules, in the presence of the decomposition $[-]$.

Before proceeding to give the logical properties of resolution proofs, we consider some examples which illustrate the more complicated aspects of the definition.

- Firstly, an example of the resolution rule:
$$\frac{c, p, p \multimap r \vdash_R r \qquad c, q_1, q_2 \vdash_R q_1 \otimes q_2}{p, q_1, q_2, !((p \multimap r) \parr ((q_1 \otimes q_2) \multimap s)) \vdash_R r, s} \tag{1}$$
where we write c for the formula $!((p \multimap r) \parr ((q_1 \otimes q_2) \multimap s))$. Note that a sequent calculus proof of this conclusion from these premisses would include occurrences of the $C!$-L, !-L, $\parr$-L and $\multimap$-L rules.

- Secondly, an example of the $\otimes$-rule:

$$\frac{!(p \maltese q), p \vdash_R p \quad !(p \maltese q), p \vdash_R p \quad !(p \maltese q), q \vdash_R q \quad !(p \maltese q), q \vdash_R q}{!(p \maltese q) \vdash_R p, p, q \otimes q}. \tag{2}$$

  Note that a sequent calculus proof of this conclusion from these premisses would include occurrences of the $C!$-L, !-L, $\otimes$-R and $\maltese$-L rules. The behaviour of the !-rule is similar to that of the $\otimes$-rule.

- Thirdly, consider the sequent

$$!(p(t) \otimes q(t)), !(\bigwedge x.((p(x) \otimes q(x)) \multimap r(x))) \vdash r(t). \tag{3}$$

  We attempt to construct a resolution proof of the sequent

$$[!(p(t) \otimes q(t)), !(\bigwedge x.((p(x) \otimes q(x)) \multimap r(x)))] \vdash_R r(t),$$

  i.e. of

$$!(p(t) \otimes q(t)), !((p(x) \otimes q(x)) \multimap r(x)) \vdash_R r(t).$$

  In this case, the taking of resolvants is rather trivial, so that applying the resolution yields the sequent

$$!(p(t) \otimes q(t)), !(\bigwedge x.((p(x) \otimes q(x)) \multimap r(x))), p(t), q(t) \vdash_R p(t) \otimes q(t).$$

  A resolution proof is then obtained by an application of the $\otimes$-rule. In § 4, we shall see that this example amounts to an example of the encoding of the usual (analytic) resolution rule of intuitionistic logic, such as that for a simple logic of Horn clauses mentioned in § 1.

- Fourthly, consider the sequent

$$\bigwedge x.(r(x) \otimes s(x) \otimes (r(x) \multimap p(x)) \otimes (s(x) \multimap q(x))) \vdash p(t) \otimes q(u), \tag{4}$$

  which is not provable in the linear sequent calculus. We attempt to construct a resolution proof of the sequent

$$r(x), s(x), r(x) \multimap p(x), s(x) \multimap q(x) \vdash_R p(t) \otimes q(u).$$

  Applying the $\otimes$-rule and the resolution rule, we obtain the two sequents

$$r(t) \vdash_R r(t) \qquad \text{and} \qquad s(u) \vdash_R s(u).$$

  Both of these are of the form of axioms: however, as the first exploits the substitution $[t/x]$ and the second exploits the substitution $[u/x]$, for the variable x, we have violated the side-condition of the definition and have failed to obtain a resolution proof.

- Finally, consider the provable sequent

$$!(\bigwedge x.p(x)) \vdash p(t) \otimes p(u). \tag{5}$$

  Since $[!(\bigwedge x.p(x))] = \{!p(x)\}$, we attempt to construct a resolution proof of the sequent

$$!p(x) \vdash_R p(t) \otimes p(u).$$

  Taking $!p(x)$-components, in order to apply the $\otimes$-rule, we must simulate a contraction on the formula $!p(x)$; but our regime of standardizing apart bound variables ensures that we obtain the sequents

$$!p(x) \vdash_R p(t) \qquad \text{and} \qquad !p(y) \vdash_R p(u).$$

  Thus we avoid violating the side-condition of the definition and are indeed able to obtain a resolution proof, as desired.

An important property of resolution proofs is the *instance property*, *i.e.* that if $\mathcal{P}[t/x] \vdash_R \mathcal{G}$, then $\mathcal{P} \vdash_R \mathcal{G}$. This is due to the encoding of the $\bigwedge$-L rule via substitutions, rather than choosing a term to which the universally quantified variable should be instantiated. For example, consider the proof in the sequent calculus of $\bigwedge x.\, p(x) \vdash p(t)$, which involves an application of $\bigwedge$-L to an initial sequent. In our case, we look for a resolution proof of $p(x) \vdash_R p(t)$, which is an initial sequent and hence provable. However, $p(t) \vdash_R p(t)$ is also provable and it is not hard to see that any term u such that t is an instance of u will be such that $p(u) \vdash_R p(t)$. Thus the use of substitutions to encode the $\bigwedge$-L rule means that we "lift" a resolution proof of $\mathcal{P}[t/x] \vdash_R \mathcal{G}$ to one of $\mathcal{P} \vdash_R \mathcal{G}$.[6]

LEMMA 3.12 (WEAKENING AND CONTRACTION) *Let $\mathcal{P}$ be a multiset of clauses, D be a clause, G be a goal formula and $\mathcal{G}$ be a multiset of goal formulae.*

1. *If $\mathcal{P} \vdash_R \mathcal{G}$, then $\mathcal{P}, !D \vdash_R \mathcal{G}$.*
2. *If $\mathcal{P} \vdash_R \mathcal{G}$, then $\mathcal{P} \vdash_R ?\, G, \mathcal{G}$.*
3. *If $\mathcal{P}, !D, !D \vdash_R \mathcal{G}$, then $\mathcal{P}, !D \vdash_R \mathcal{G}$.*
4. *If $\mathcal{P} \vdash_R ?G, ?G, \mathcal{G}$, then $\mathcal{P} \vdash_R ?\, G, \mathcal{G}$.*
5. *If $\mathcal{P} \cup \{D\}^n \vdash_R \mathcal{G}$ for some integer $n \geq 0$ then $\mathcal{P}, !D \vdash_R \mathcal{G}$.*
6. *If $\mathcal{P} \vdash_R \{G\}^n, \mathcal{G}$ for some $n \geq 0$, then $\mathcal{P} \vdash_R ?\, G, \mathcal{G}$.*

PROOF The proof of 6 is immediate by an application of the ?-rule. The others can be shown by induction on the height of the resolution proof. As these proofs are straightforward and repetitive, we sketch the proof for 1 and omit the others.

The base case occurs when $\mathcal{P} \vdash_R G$ is initial; it is clear that the lemma holds in this case. Hence we assume that the lemma is true for all sequents whose proof is no more than a given height.

Consider the step used to derive the current sequent: we present the argument only for the cases for the resolution rule and the $\otimes$-rule.

resolution: The previous sequents must be given by $\mathcal{D}_1 \vdash_R \mathcal{G}_1 \ldots \mathcal{D}_n \vdash_R \mathcal{G}_n$, where the multiset $\bigcup_{i=1}^n \{\mathcal{D}_i \vdash_R \mathcal{G}_i\}$ is a resolvant of $\mathcal{P} \vdash_R \mathcal{G}$; so by the hypothesis we have $\mathcal{D}_1, !D \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n, !D \vdash_R \mathcal{G}_n$; as this is (clearly) a resolvant of $\mathcal{P}, !D \vdash_R \mathcal{G}$, we can apply resolution to obtain $\mathcal{P}, !D \vdash_R \mathcal{G}$.

$\otimes$-rule: The previous sequents must be given by $\mathcal{D}_0 \vdash_R G_1, \mathcal{G}_0$, $\mathcal{D}'_0 \vdash_R G_2, \mathcal{G}'_0$ and $\mathcal{D}_1 \vdash_R \mathcal{G}_1 \ldots \mathcal{D}_n \vdash_R \mathcal{G}_n$, where $\{\mathcal{D}_0, \mathcal{D}'_0 \vdash_R G_1 \otimes G_2, \mathcal{G}_0, \mathcal{G}'_0\} \cup \bigcup_{i=1}^n \{\mathcal{D}_i \vdash_R \mathcal{G}_i\}$ is a multiset of components of $\mathcal{P} \vdash_R G_1 \otimes G_2, \mathcal{G}'$; so by the hypothesis we have $\mathcal{D}_0, !D \vdash_R G_1, \mathcal{G}_0$, $\mathcal{D}'_0, !D \vdash_R G_2, \mathcal{G}'_0$, $\mathcal{D}_1, !D \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n, !D \vdash_R \mathcal{G}_n$; as this is (clearly) a multiset of components of $\mathcal{P}, !D \vdash_R G_1 \otimes G_2 \mathcal{G}'$, we can apply the $\otimes$-rule to obtain $\mathcal{P}, !D \vdash_R G_1 \otimes G_2, \mathcal{G}'$.

The other cases are similar.

Next we show a lemma of some technical importance; essentially, this is that an occurrence of the resolution rule using a clause C can be replaced by an occurrence of the resolution rule on a clause $!C$ in a corresponding program. This correspondence is given by the mapping $-^!$, defined below.

DEFINITION 3.13 (THE MAPPING $-^!$) Let $\mathcal{D}$ be a multiset of clauses. We define a mapping $-^!$ from multisets of clauses to multisets of clauses as follows:

$$\mathcal{D}^! = \{\, ! \bigotimes_{C \in [\mathcal{D}]^L} C \,\} \cup \mathcal{D}^E \ .$$

Note the similarity between this definition and the definition of $[!D]$. We can think of the mapping $-^!$ as converting a set of linear clauses into a single clause. Below we show that this mapping preserves provability via resolution proofs.

[6] For similar reasons, the provability of $\mathcal{P} \vdash_R \mathcal{G}$ implies the provability of $\mathcal{P} \vdash_R \mathcal{G}[t/x]$ where x is free in $\mathcal{G}$.

LEMMA 3.14 (SOUNDNESS OF $-^!$) *Let $\mathcal{P}$ and $\mathcal{D}$ be multisets of clauses and let $\mathcal{G}$ be a multiset of goal formulae. If $\mathcal{P}, \mathcal{D} \vdash_R \mathcal{G}$ then $\mathcal{P}, \mathcal{D}^! \vdash_R \mathcal{G}$.*

PROOF We proceed by induction on the height of the proof. The base case occurs when $\mathcal{P}, \mathcal{D} \vdash_R \mathcal{G}$ is initial; it is clear that the lemma holds in this case.

Consider the step used to derive the current sequent: again we present the argument for just the resolution rule and the $\otimes$-rule.

resolution: The previous sequents must be given by $\mathcal{D}_1 \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n \vdash_R \mathcal{G}_n$, where $\bigcup_{i=1}^n \{\mathcal{D}_i \vdash_R \mathcal{G}_i\}$ is a resolvant of $\mathcal{P}, D, \vdash_R \mathcal{G}$; by Lemma 3.12, we have that $\mathcal{D}_1, D^! \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n, D^! \vdash_R \mathcal{G}_n$; as this is a resolvant of $\mathcal{P}, D^! \vdash_R \mathcal{G}$, we have, by an application of the resolution rule, $\mathcal{P}, D^! \vdash_R \mathcal{G}$.

$\otimes$-rule: The previous sequents must be given by $\mathcal{D}_0 \vdash_R G_1, \mathcal{G}_0$, $\mathcal{D}'_0 \vdash_R G_2, \mathcal{G}'_0$ and $\mathcal{D}_1 \vdash_R \mathcal{G}_1 \ldots \mathcal{D}_n \vdash_R \mathcal{G}_n$, where $\{\mathcal{D}_0, \mathcal{D}'_0 \vdash_R G_1 \otimes G_2, \mathcal{G}_0, \mathcal{G}'_0\} \cup \bigcup_{i=1}^n \{\mathcal{D}_i \vdash_R \mathcal{G}_i\}$ is a multiset of components of $\mathcal{P}, D \vdash_R G_1 \otimes G_2, \mathcal{G}'$; by Lemma 3.12, we have that $\mathcal{D}_0, D^! \vdash_R G_1, \mathcal{G}_0$, $\mathcal{D}'_0, D^! \vdash_R G_2, \mathcal{G}'_0$, $\mathcal{D}_1, D^! \vdash_R \mathcal{G}_1, \ldots, \mathcal{D}_n, D^! \vdash_R \mathcal{G}_n$; as this is a multiset of components of $\mathcal{P}, D^! \vdash_R G_1 \otimes G_2, \mathcal{G}'$, we can apply the $\otimes$-rule to obtain $\mathcal{P}, D^! \vdash_R G_1 \otimes G_2, \mathcal{G}'$.

The other cases are similar.

PROPOSITION 3.15 (COMPONENTS) *Let $\mathcal{P}$ be a multiset of clauses and let $\mathcal{G}$ be a multiset of goal formulae. Then $\mathcal{P} \vdash_R \mathcal{G}$ iff there is a multiset of components $\mathcal{C}$ of $\mathcal{P} \vdash_R \mathcal{G}$ such that each element of $\mathcal{C}$ has a resolution proof.*

PROOF The forwards direction is obvious. For the backwards direction, we proceed by induction on the height of the shortest resolution proof of the individual components.

In the base case, one of the components is an initial sequent, so the multiset of components is a resolvant of $\mathcal{P} \vdash_R \mathcal{G}$; so by the resolution rule we have a resolution proof of $\mathcal{P} \vdash_R \mathcal{G}$.

Hence we assume that the proposition is true for all multisets of components of $\mathcal{P} \vdash_R \mathcal{G}$ for which each element has a resolution proof and that the shortest such resolution proof is of no more than a given height.

Let $\mathcal{C}$ be $\{\mathcal{P}_1 \vdash_R \mathcal{G}_1\} \cup \ldots \cup \{\mathcal{P}_n \vdash_R \mathcal{G}_n\}$ and for simplicity assume that the sequent with the shortest resolution proof is $\mathcal{P}_1 \vdash_R \mathcal{G}_1$.

Consider the rule used to derive $\mathcal{P}_1 \vdash_R \mathcal{G}_1$. There are twelve cases, of which we give just a few.

resolution: In this case we have that the sequent $\mathcal{P}'_i \vdash_R \mathcal{G}'_i$ has a resolution proof for each $1 \leq i \leq m$, where $\{\mathcal{P}'_1 \vdash_R \mathcal{G}'_1\} \cup \ldots \cup \{\mathcal{P}'_n \vdash_R \mathcal{G}'_m\}$ is a resolvant of $\mathcal{P}_1 \vdash_R \mathcal{G}_1$. Now, as $\bigcup_{i=1}^n \{\mathcal{P}_i \vdash_R \mathcal{G}_i\}$ is a multiset of components of $\mathcal{P} \vdash_R \mathcal{G}$, we have that $\bigcup_{i=1}^m \{\mathcal{P}'_i \vdash_R \mathcal{G}'_i\} \cup \bigcup_{i=2}^n \{\mathcal{P}_i \vdash_R \mathcal{G}_i\}$ is a resolvant of $\mathcal{P} \vdash_R \mathcal{G}$ and, by the hypothesis, each element of the resolvant is provable, so that by an application of the resolution rule, we have that $\mathcal{P} \vdash_R \mathcal{G}$ has a resolution proof.

& -rule: In this case we have that $\mathcal{P}'_1 \vdash_R G'_1, \mathcal{G}'$ and $\mathcal{P}''_1 \vdash_R G''_2, \mathcal{G}''$, where $\mathcal{G}_1 = G_1 \,\&\, G_2, \mathcal{G}_0$, $\mathcal{P}'_1$ and $\mathcal{P}''_1$ are distinct copies of $\mathcal{P}_1$ and G'_1, G''_2, $\mathcal{G}'$ and $\mathcal{G}''$ are the correspondingly updated versions of G_1, G_2 and $\mathcal{G}_0$ respectively. Now if we let $\mathcal{P}'_i$ be a distinct copy of $\mathcal{P}_i$ and $\mathcal{G}'_i$ be the correspondingly updated version of $\mathcal{G}_i$, for $2 \leq i \leq n$, then we have that $\{\mathcal{P}'_1 \vdash_R G'_1, \mathcal{G}'\} \cup \bigcup_{i=2}^n \{\mathcal{P}'_i \vdash_R \mathcal{G}'_i\}$ is a multiset of components of $\mathcal{P}' \vdash_R G'_1, \mathcal{G}'$, where $\mathcal{P}' \vdash_R G'_1, \mathcal{G}'$ is a distinct copy of $\mathcal{P} \vdash_R G_1, \mathcal{G}_0$; so by the hypothesis, we have that $\mathcal{P}' \vdash_R G'_1, \mathcal{G}'$ has a resolution proof. By a similar argument, we have that $\mathcal{P}'' \vdash_R G''_2, \mathcal{G}''$ has a resolution proof, as does $\mathcal{P} \vdash_R G_1 \,\&\, G_2, \mathcal{G}'$.

$\otimes$-rule: In this case we have that the sequents $\mathcal{P}_0 \vdash_R G_1, \mathcal{G}_0$, $\mathcal{P}'_0 \vdash_R G_2, \mathcal{G}'_0$ and each $\mathcal{P}'_i \vdash_R \mathcal{G}'_i$ have resolution proofs, for $1 \leq i \leq m$, where
$\{\mathcal{P}_0, \mathcal{P}'_0 \vdash_R G_1 \otimes G_2, \mathcal{G}_0, \mathcal{G}'_0\} \cup \{\mathcal{P}'_1 \vdash_R \mathcal{G}'^{\prime}_1\} \cup \ldots \cup \{\mathcal{P}'_m \vdash_R \mathcal{G}'_m\}$ is a multiset of components of $\mathcal{P}_1 \vdash_R \mathcal{G}_1$. We have that $\{\mathcal{P}_0, \mathcal{P}'_0 \vdash_R G_1 \otimes G_2, \mathcal{G}_0, \mathcal{G}'_0\} \cup$
$\{\mathcal{P}'_1 \vdash_R \mathcal{G}'^{\prime}_1\} \cup \ldots \cup \{\mathcal{P}'_m \vdash_R \mathcal{G}'_m\} \cup \bigcup_{i=2}^n \{\mathcal{P}_i \vdash_R \mathcal{G}_i\}$ is a multiset of components of $\mathcal{P} \vdash_R G_1 \otimes G_2, \mathcal{G}'$, so by the hypothesis we have that $\mathcal{P} \vdash_R G_1 \otimes G_2, \mathcal{G}'$ has a resolution proof.

!-rule: In this case we have that the sequents $\mathcal{P}_0 \vdash_R G, \mathcal{G}_0$ and each $\mathcal{P}'_i \vdash_R \mathcal{G}'_i$ have resolution proofs, for $1 \leq i \leq m$, where $\{\mathcal{P}_0 \vdash_R G, \mathcal{G}_0\} \cup \{\mathcal{P}'_1 \vdash_R \mathcal{G}'_1\} \cup \ldots \cup \{\mathcal{P}'_n \vdash_R \mathcal{G}'_m\}$ is a multiset of components of $\mathcal{P}_1 \vdash_R \mathcal{G}_1$ and $\mathcal{P}_0^L = \emptyset = \mathcal{G}_0^L$. Now, as $\bigcup_{i=1}^{n}\{\mathcal{P}_i \vdash_R \mathcal{G}_i\}$ is a multiset of components of the sequent $\mathcal{P} \vdash_R \mathcal{G}$, we have that $\{\mathcal{P}_0 \vdash_R G, \mathcal{G}_0\} \cup \bigcup_{i=1}^{m}\{\mathcal{P}'_i \vdash_R \mathcal{G}'_i\} \cup \bigcup_{i=2}^{n}\{\mathcal{P}_i \vdash_R \mathcal{G}_i\}$ is a multiset of components of the sequent $\mathcal{P} \vdash_R \mathcal{G}$; so by the hypothesis we have that $\mathcal{P} \vdash_R \mathcal{G}$ has a resolution proof.

The other cases are similar.

PROPOSITION 3.16 (SOUNDNESS OF RESOLUTION PROOFS) *Let $\mathcal{P}$ be a multiset of definite formulae and let $\mathcal{G}$ be a multiset of goal formulae. If $[\mathcal{P}] \vdash_R \mathcal{G}$, then $\mathcal{P} \vdash \mathcal{G}$ has a uniform (simple locally LR) proof.*

PROOF By Theorem 2.16, it will suffice to show $\mathcal{P} \vdash \mathcal{G}$ is provable.

We proceed by induction on the size of the proof. The base case occurs when one of the following conditions holds:

- $\mathcal{P}^L = \{A'\}$, $\mathcal{G}^L = \{A\}$ and $A = A'[\vec{t}/\vec{x}]$;
- $\mathcal{P}^L = \emptyset$, $\mathcal{G}^L = \{A\}$ and there exists $!A' \in \mathcal{P}^E$ such that $A = A'[\vec{t}/\vec{x}]$;
- $\mathcal{P}^L = \emptyset$ and $\mathcal{G}^L = \{\,\mathbf{1}\,\}$;
- $\mathcal{P}^L = \{\bot\}$ and $\mathcal{G}^L = \emptyset$;
- $\top \in \mathcal{G}$.

It should be clear that $\mathcal{P} \vdash \mathcal{G}$ is provable in each case.

Hence the induction hypothesis is that the proposition holds for all sequents whose extended resolution proofs no more than a given height. Consider the step used to derive the current sequent. There are twelve cases, many of which are similar; so we present the argument for just some of the cases.

resolution: In this case we have that for some resolvant $\bigcup_{i=1}^{n}\{\mathcal{P}_i \vdash_R \mathcal{G}_i\}$ of $\mathcal{P} \vdash \mathcal{G}$ that each $\mathcal{P}_i \vdash_R \mathcal{G}_i$ has a resolution proof. By the hypothesis we have that each $\mathcal{D}_i \vdash \mathcal{G}_i$ has a linear proof, where $[\mathcal{D}_i] = \mathcal{P}_i$. Now clearly the reconstruction of $\mathcal{D}_i$ from $\mathcal{P}_i$ involves the insertion of a sequence of left rules other than $\Cross$-L. We can then apply $\Cross$-L to reduce the number of components and then insert a sequence of left rules, possibly including $\Cross$-L, into the proof to get a linear proof of $\mathcal{P} \vdash \mathcal{G}$.

&-rule: In this case we have that $\mathcal{P}_1 \vdash_R G'_1, \mathcal{G}''$ and $\mathcal{P}_2 \vdash_R G'_2, \mathcal{G}'''$, where $\mathcal{G} = \{G_1 \,\&\, G_2\} \cup \mathcal{G}'$, $\mathcal{P}_1$, $\mathcal{P}_2$ are distinct copies of $\mathcal{P}$ and G'_1, G'_2, $\mathcal{G}''$ and $\mathcal{G}'''$ are the correspondingly updated versions of G_1, G_2 and $\mathcal{G}'$ respectively. So by the hypothesis, $\mathcal{P}'_1 \vdash G'_1, \mathcal{G}''$ and $\mathcal{P}'_2 \vdash G'_2, \mathcal{G}'''$ are provable, where $\mathcal{P}'_1$ and $\mathcal{P}'_2$ are distinct copies of $\mathcal{P}$. It follows that $\mathcal{P} \vdash G_1, \mathcal{G}'$ and $\mathcal{P} \vdash G_2, \mathcal{G}'$ are provable; so $\mathcal{P} \vdash G_1 \,\&\, G_2, \mathcal{G}'$ is provable.

$\otimes$-rule: In this case we have that $\mathcal{P}_0 \vdash_R G_1, G_0$, $\mathcal{P}'_0 \vdash_R G_2, \mathcal{G}'_0$ and each $\mathcal{P}_i \vdash_R \mathcal{G}_i$ have resolution proofs, where $\{\mathcal{P}_0, \mathcal{P}'_0 \vdash_R G_1 \otimes G_2, \mathcal{G}_0, \mathcal{G}'_0\} \cup \bigcup_{i=1}^{n}\{\mathcal{P}_i \vdash_R \mathcal{G}_i\}$ is a multiset of components of $\mathcal{P} \vdash_R G_1 \otimes G_2, \mathcal{G}'$ and $\mathcal{G} = G_1 \otimes G_2, \mathcal{G}'$. By the hypothesis, we have that $\mathcal{D}_0 \vdash G_1, G_0$, $\mathcal{D}'_0 \vdash G_2, \mathcal{G}'_0$ and each $\mathcal{D}_i \vdash \mathcal{G}_i$ has a linear proof, where $[\mathcal{D}_0] = \mathcal{P}_0$, $[\mathcal{D}'_0] = \mathcal{P}'_0$ and $[\mathcal{D}_i] = \mathcal{P}_i$. Clearly then the sequent $\mathcal{D}_0, \mathcal{D}'_0 \vdash G_1 \otimes G_2, \mathcal{G}_0, \mathcal{G}'_0$ is provable; we can then apply $\Cross$-L to reduce the number of components and then insert a sequence of left rules, possibly including $\Cross$-L, into the proof to get a proof of $\mathcal{P} \vdash G_1 \otimes G_2, \mathcal{G}'$.

!-rule: In this case we have that the sequent $\mathcal{P}_0 \vdash_R G, G_0$ and each $\mathcal{P}_i \vdash_R \mathcal{G}_i$ have resolution proofs, where $\{\mathcal{P}_0 \vdash_R !G, \mathcal{G}_0\} \cup \bigcup_{i=1}^{n}\{\mathcal{P}_i \vdash_R \mathcal{G}_i\}$ is a multiset of components of $\mathcal{P} \vdash_R !G, \mathcal{G}'$, $\mathcal{G} = !G, \mathcal{G}'$ and $\mathcal{P}_0^L = \emptyset = \mathcal{G}_0^L$. By the hypothesis, we have that $\mathcal{D}_0 \vdash !G, G_0$ and each $\mathcal{D}_i \vdash \mathcal{G}_i$ has a proof, where $[\mathcal{D}_i] = \mathcal{P}_i$, for $0 \leq i \leq n$. Now we can reconstruct $\mathcal{D}_0$ from $\mathcal{P}_0$ in such a way that $\mathcal{D}_0^L = \emptyset$; so we can apply the !-R rule to get a proof of $\mathcal{D}_0 \vdash !G, \mathcal{G}_0$. We can then apply $\Cross$-L to reduce the number of components and then insert a sequence of left rules, possibly including $\Cross$-L, into the proof to get a proof of $\mathcal{P} \vdash !G, \mathcal{G}'$.

We remark that for this fragment of linear logic we have, by Theorem 2.16, that for any linear proof there is a corresponding locally LR proof.

PROPOSITION 3.17 (COMPLETENESS OF RESOLUTION PROOFS) *Let $\mathcal{P}$ be a multiset of definite formulae and let $\mathcal{G}$ be a multiset of goal formulae. If $\mathcal{P} \vdash \mathcal{G}$ has a uniform (simple locally LR) proof, then $[\mathcal{P}] \vdash_R \mathcal{G}$.*

PROOF We proceed by induction on the length of the proof of $\mathcal{P} \vdash \mathcal{G}$. The base case occurs when $\mathcal{P} \vdash \mathcal{G}$ is initial. There are four cases:

axiom: $\mathcal{P} = \mathcal{G} = \{A\}$ for some atom A;

$\mathrm{L}^{\perp}$: $\mathcal{P} = \{\perp\}$ and $\mathcal{G} = \emptyset$;

1-R: $\mathcal{P} = \emptyset$ and $\mathcal{G} = \{\mathbf{1}\}$;

$\top$-R: $\top \in \mathcal{G}$.

Clearly the proposition holds in each of these cases.

Hence we assume that the proposition holds for all sequents whose proof is of no more than a given height. Consider the rule used to derive $\mathcal{P} \vdash \mathcal{G}$. By Lemma 2.15, we need only consider the rules **1**-L, $\mathrm{L}^{\perp}$, $\with$-L, $\otimes$-L, $\parr$-L, $\bigwedge$-L, $\multimap$-L, !-L, C!-L, W!-L together with the right rules (note that $\mathrm{L}^{\perp}$, **1**-R and $\top$-R are covered by the base case). As many of these are similar, we present the argument just for selected cases.

$\with$-L: The previous sequents must be $D_1, \mathcal{P}_1 \vdash \mathcal{G}_1$ and $D_2, \mathcal{P}_2 \vdash \mathcal{G}_2$, where $\{D_1 \with D_2\} \cup \mathcal{P}_1 \cup \mathcal{P}_2 = \mathcal{P}$ and $\mathcal{G}_1 \cup \mathcal{G}_2 = \mathcal{G}$. So by the hypothesis, $[D_1], [\mathcal{P}_1] \vdash_R \mathcal{G}_1$ and $[D_2], [\mathcal{P}_2] \vdash_R \mathcal{G}_2$, so by Proposition 3.15, $[\mathcal{P}] \vdash_R \mathcal{G}$.

& -L: The previous sequent must be either $D_1, \mathcal{P}' \vdash \mathcal{G}$ or $D_2, \mathcal{P}' \vdash \mathcal{G}$, where $\{D_1 \mathbin{\&} D_2\} \cup \mathcal{P}' = \mathcal{P}$. So by the hypothesis, either $[D_1], [\mathcal{P}'] \vdash_R \mathcal{G}$ or $[D_2], [\mathcal{P}'] \vdash_R \mathcal{G}$; as either of these sequents is a multiset of components of $[\mathcal{P}] \vdash_R \mathcal{G}$, by Proposition 3.15, $[\mathcal{P}] \vdash_R \mathcal{G}$.

$\bigwedge$-L: The previous sequent must be $D[t/x], \mathcal{P}' \vdash \mathcal{G}$, where $\{\bigwedge x.D\} \cup \mathcal{P}' = \mathcal{P}$. So by the hypothesis, we have that $[D[t/x], \mathcal{P}'] \vdash_R \mathcal{G}$, *i.e.* $[D[t/x]] \cup [\mathcal{P}'] \vdash_R \mathcal{G}$. Hence by the instance property, $[D] \cup [\mathcal{P}'] \vdash_R \mathcal{G}$ (the resolution proof of the induction hypothesis will suffice) and so $[\bigwedge x.D, \mathcal{P}'] \vdash_R \mathcal{G}$.

$\multimap$-L: The previous sequent must be $\mathcal{P}' \vdash G, \mathcal{G}'$ where $\{G \multimap A\} \cup \mathcal{P}' = \mathcal{P}$ and $\{A\} \cup \mathcal{G}' = \mathcal{G}$, so by the hypothesis, $[\mathcal{P}'] \vdash_R G, \mathcal{G}'$ and, as this is clearly a resolvant of $[\mathcal{P}] \vdash_R \mathcal{G}$, we can apply the resolution rule to obtain a resolution proof of $[\mathcal{P}] \vdash_R \mathcal{G}$.

!-L: The previous sequent must be $\mathcal{P}', D \vdash G, \mathcal{G}'$, where $\{!D\} \cup \mathcal{P}' = \mathcal{P}$. So by the hypothesis, $[\mathcal{P}'], [D] \vdash_R G, \mathcal{G}'$; then by Lemma 3.14, $[\mathcal{P}'], [D]^{!} \vdash_R G, \mathcal{G}'$, *i.e.* $[\mathcal{P}] \vdash_R \mathcal{G}$.

$\otimes$-R: The previous sequents must be $\mathcal{P}_1 \vdash G_1, \mathcal{G}_1$ and $\mathcal{P}_2 \vdash G_2, \mathcal{G}_2$, where we have $\mathcal{P}_1 \cup \mathcal{P}_2 = \mathcal{P}$ and $\{G_1 \otimes G_2\}, \mathcal{G}_1 \cup \mathcal{G}_2 = \mathcal{G}$. So by the hypothesis, $[\mathcal{P}_1] \vdash_R G_1, \mathcal{G}_1$ and $[\mathcal{P}_2] \vdash_R G_2, \mathcal{G}_2$. As these two sequents form a multiset of components of $[\mathcal{P}_1], [\mathcal{P}_2] \vdash_R G_1 \otimes G_2, \mathcal{G}_1, \mathcal{G}_2$, we can apply the $\otimes$-rule to get a resolution proof of $[\mathcal{P}_1], [\mathcal{P}_2] \vdash_R G_1 \otimes G_2, \mathcal{G}_1, \mathcal{G}_2$; then by Lemma 3.12 we have that $[\mathcal{P}] \vdash_R \mathcal{G}$ has a resolution proof.

& -R: The previous sequents must be $\mathcal{P} \vdash G_1, \mathcal{G}'$ and $\mathcal{P} \vdash G_2, \mathcal{G}'$, where $\{G_1 \mathbin{\&} G_2\} \cup \mathcal{G}' = \mathcal{G}$. So the sequents $\mathcal{P}' \vdash G_1', \mathcal{G}''$ and $\mathcal{P}'' \vdash G_2', \mathcal{G}'''$ are provable, where $\mathcal{P}'$ and $\mathcal{P}''$ are distinct copies of $\mathcal{P}$ and G_1', G_2', $\mathcal{G}''$ and $\mathcal{G}'''$ are the correspondingly updated versions of G_1, G_2 and $\mathcal{G}'$ respectively. So by the hypothesis, we have that $[\mathcal{P}'] \vdash_R G_1', \mathcal{G}''$ and $[\mathcal{P}''] \vdash_R G_2', \mathcal{G}'''$, *i.e.* $[\mathcal{P}] \vdash_R \mathcal{G}$.

?-R: The previous sequent must be $\mathcal{P} \vdash G, \mathcal{G}'$, where $\{?\,G\} \cup \mathcal{G}' = \mathcal{G}$. So by the hypothesis, we have that $[\mathcal{P}] \vdash_R G, \mathcal{G}'$; hence by Lemma 3.12, $[\mathcal{P}] \vdash_R G, ?\,G, \mathcal{G}'$ *i.e.* $[\mathcal{P}] \vdash_R \mathcal{G}$.

!-R: The previous sequent must be $!\,\mathcal{P} \vdash G, ?\,\mathcal{G}'$, where $\{!\,G\} \cup ?\,\mathcal{G}' = \mathcal{G}$. So by the hypothesis, we have that $[!\,\mathcal{P}] \vdash_R G, ?\,\mathcal{G}'$; clearly, we then have $[!\,\mathcal{P}] \vdash_R \mathcal{G}$.

In summary, we have the following:

THEOREM 3.18 *Let $\mathcal{P}$ be a multiset of definite formulae and let $\mathcal{G}$ be a multiset of goal formulae. Then $\mathcal{P} \vdash \mathcal{G}$ is provable iff $[\mathcal{P}] \vdash_R \mathcal{G}$.*

4 Intuitionistic Logic

As noted above, the presence of the exponential, !, means that the rules of weakening and contraction can be re-introduced in a controlled manner. Hence it is natural to ask what is the shape of purely intuitionistic deduction in this setting.[7] In particular, we show how intuitionistic proofs in the hereditary Harrop formulae fragment of intuitionistic first-order logic can be encoded as proofs in this fragment of linear logic.

In order so to do, we recall some definitions from [19, 17] for intuitionistic logic. We assume that we are dealing with first-order intuitionistic logic and first-order linear logic over some common language of constants and variables and function and predicate symbols.

DEFINITION 4.1 [Hereditary Harrop formulae] Let A range over atomic formulae. We define the classes of *hereditary Harrop definite formulae* and *hereditary Harrop goal formulae* as follows:

$$\begin{array}{lrcl} \textit{Definite formulae} & D & ::= & A \mid D \wedge D \mid G \supset A \mid \forall x.D \\ \textit{Goal formulae} & G & ::= & A \mid G \vee G \mid G \wedge G \mid D \supset G \mid \forall x.G \mid \exists x.G. \end{array}$$

An *hereditary Harrop program* is a set of closed hereditary Harrop definite formulae. An *hereditary Harrop goal* is a closed Hereditary Harrop goal formula.

Now we recall the definition of the relation $\vdash_O$ [19, 17]. In order to avoid notational confusion, we use $\|-\|$ to denote the mapping $[-]$ of [17].

DEFINITION 4.2 [Hereditary Harrop Clausal Decomposition] Let $\cup$ denote set union. We define a mapping $\|-\|$ from sets of hereditary Harrop definite formulae to sets of hereditary Harrop definite formulae inductively as follows:

$$\begin{array}{lll} \|\mathcal{P}\| & =_{\rm def} & \bigcup_{D \in \mathcal{P}} \|D\| \\ \|A\| & =_{\rm def} & \{ A \} \\ \|D_1 \wedge D_2\| & =_{\rm def} & \|D_1\| \cup \|D_2\| \\ \|G \supset A\| & =_{\rm def} & \{ G \supset A \} \\ \|\forall x.D\| & =_{\rm def} & \bigcup_{t \in \mathcal{U}} \|D[t/x]\|, \end{array}$$

where $\mathcal{U}$ denotes the universe of terms.

DEFINITION 4.3 [Hereditary Harrop Uniform Proofs] Let $\mathcal{P}$ be a hereditary Harrop program and let $\mathcal{G}$ be a hereditary Harrop goal. The relation $\vdash_O$ is the least relation satisfying:

- $\mathcal{P} \vdash_O A$ iff either $A \in \|\mathcal{P}\|$ or there is some $G \supset A \in \|\mathcal{P}\|$ such that $\mathcal{P} \vdash_O G$;
- $\mathcal{P} \vdash_O G_1 \vee G_2$ iff $\mathcal{P} \vdash_O G_1$ or $\mathcal{P} \vdash_O G_2$;
- $\mathcal{P} \vdash_O G_1 \wedge G_2$ iff $\mathcal{P} \vdash_O G_1$ and $\mathcal{P} \vdash_O G_2$;
- $\mathcal{P} \vdash_O D \supset G$ iff $\mathcal{P}, D \vdash_O G$;
- $\mathcal{P} \vdash_O \exists x.G$ iff $\mathcal{P} \vdash_O G[t/x]$ for some $t \in \mathcal{U}$;
- $\mathcal{P} \vdash_O \forall x.G$ iff $\mathcal{P} \vdash_O G[y/x]$ where y is not free in $\mathcal{P}$.

The following result was shown in [19]:

THEOREM 4.4 (INTUITIONISTIC OPERATIONAL PROVABILITY) *Let $\mathcal{P}$ be a hereditary Harrop formula program and let G be a hereditary Harrop goal. Let $\vdash_I$ denote intuitionistic provability. Then $\mathcal{P} \vdash_O G$ iff $\mathcal{P} \vdash_I G$.*

Next we present the encoding of intuitionistic formulae in linear formulae, via two mappings, $(-)^+$ and $(-)^-$.

[7] In fact, for simplicity, we treat here just minimal deduction.

DEFINITION 4.5 [Encoding] Let D range over hereditary Harrop definite formulae, let $\mathcal{D}$ range over multisets of (linear) definite formulae and let G range over hereditary Harrop goals. We define the linear encoding $(\mathcal{D})^-$ of $\mathcal{D}$ and the linear encoding $(G)^+$ of G as follows:

$$
\begin{array}{lll@{\qquad}lll}
(A)^+ & =_{\mathrm{def}} & A & & & \\
(G_1 \wedge G_2)^+ & =_{\mathrm{def}} & (G_1)^+ \,\&\, (G_2)^+ & \mathcal{D}^- & =_{\mathrm{def}} & \bigcup_{D \in \mathcal{D}} (D)^- \\
(G_1 \vee G_2)^+ & =_{\mathrm{def}} & (G_1)^+ \oplus (G_2)^+ & (D)^- & =_{\mathrm{def}} & \bigcup_{C \in \|D\|} \{(C)^-\} \\
(\exists \vec{x}.G)^+ & =_{\mathrm{def}} & \bigvee \vec{x}\,.\,(G)^+ & (\forall \vec{x}.A)^- & =_{\mathrm{def}} & !(\bigwedge \vec{x}\,.\,A) \\
(\forall \vec{x}.G)^+ & =_{\mathrm{def}} & \bigwedge \vec{x}\,.\,(G)^+ & (\forall \vec{x}.G \supset A)^- & =_{\mathrm{def}} & !(\bigwedge \vec{x}\,.\,(G)^+ \multimap A), \\
(D \supset G)^+ & =_{\mathrm{def}} & (\bigotimes (D)^-) \multimap (G)^+ & & &
\end{array}
$$

where $\bigotimes \mathcal{D}$ denotes $\bigotimes_{C \in \mathcal{D}} C$.

It should be clear that if G is a hereditary Harrop goal then $(G)^+$ is a goal formula and if D is hereditary Harrop definite formula then $(D)^-$ is a definite formula. Note that, in order to be able to define this translation, we require just single-conclusioned linear sequents, without $\Yinyang$ in definite formulae and without the exponentials in goal formulae. In such circumstances, the analysis of § 3 can be simplified considerably, *q.v.* [11], [10], [21].

We can think of these encodings as operating at the level of proofs, rather than at the level of formulae. This is due to the way that intuitionistic conjunctions and implications are translated differently, depending on whether they occur in a positive or negative position. We remark that a translation in which both occurrences of the intuitionistic conjunction are mapped to $\otimes$ is possible [18], but that this translation does not seem to work as well at the level of proofs.

We now show that this encoding behaves as expected. Note that although the proof of Proposition 4.6 is by induction over the structure of proofs, it is driven by the structure of goal formulae, with all of the left inferences being concentrated in the atomic case. It seems to be that such an analysis is possible only for uniform proofs.

PROPOSITION 4.6 (RESOLUTION PROVABILITY UNDER ENCODING) *Let $\mathcal{P}$ be a hereditary Harrop program and let G be a hereditary Harrop goal formula. Then $\mathcal{P} \vdash_O G$ iff $(\mathcal{P})^- \vdash_R (G)^+$.*

PROOF We proceed by induction on the size of the proofs. The base case is when G is an atom; it is clear that the proposition holds in this case.

The induction hypothesis is that the proposition holds for all proofs of no more than a given size. There are six cases.

A: $\mathcal{P} \vdash_O A$ iff either $A \in \|\mathcal{P}\|$, in which case we are done (this is the base of the induction), or there is some $G \supset A \in \|\mathcal{P}\|$ such that $\mathcal{P} \vdash_O G$; by the hypothesis this is equivalent to $[(\mathcal{P})^-] \vdash_R (G)^+$. This in turn is equivalent to the existence of a clause $!(G' \multimap A') \in [(\mathcal{P})^-]$ such that $(G)^+ \multimap A = (G' \multimap A')[\vec{t}/\vec{x}]$, for some $\vec{t}$, and $[(\mathcal{P})^-] \vdash_R (G)^+$, which is just the sequent $[(\mathcal{P})^-] \vdash_R (A)^+$.

$G_1 \vee G_2$: $\mathcal{P} \vdash_O G_1 \vee G_2$ iff $\mathcal{P} \vdash_O G_1$ or $\mathcal{P} \vdash_O G_2$ and by the hypothesis this is equivalent to either $[(\mathcal{P})^-] \vdash_R (G_1)^+$ or $[(\mathcal{P})^-] \vdash_R (G_2)^+$, which in turn is equivalent to the sequent $[(\mathcal{P})^-] \vdash_R (G_1)^+ \oplus (G_2)^+$, which is just $[(\mathcal{P})^-] \vdash_R (G_1 \vee G_2)^+$.

$G_1 \wedge G_2$: $\mathcal{P} \vdash_O G_1 \wedge G_2$ iff $\mathcal{P} \vdash_O G_1$ and $\mathcal{P} \vdash_O G_2$ and by the hypothesis this is equivalent to the sequents $[(\mathcal{P})^-] \vdash_R (G_1)^+$ and $[(\mathcal{P})^-] \vdash_R (G_2)^+$, which in turn is equivalent to having $[(\mathcal{P})^-] \vdash_R (G_1)^+ \,\&\, (G_2)^+$ (by some renaming), which is just the sequent $[(\mathcal{P})^-] \vdash_R (G_1 \wedge G_2)^+$.

$\exists x.G$: $\mathcal{P} \vdash_O \exists x.G$ iff $\mathcal{P} \vdash_O G[t/x]$ for some $t \in \mathcal{U}$; by the hypothesis this is equivalent to the sequent $[(\mathcal{P})^-] \vdash_R (G[t/x])^+$, which in turn is equivalent to the sequent $[(\mathcal{P})^-] \vdash_R \bigvee x\,.\,(G)^+$, which is just the sequent $[(\mathcal{P})^-] \vdash_R (\exists x.G)^+$.

$\forall x.G$: $\mathcal{P} \vdash_O \forall x.G$ iff $\mathcal{P} \vdash_O G[y/x]$, where y does not occur free in P; by the hypothesis this is equivalent to the sequent $[(\mathcal{P})^-] \vdash_R (G[y/x])^+$, which in turn is equivalent to $[(\mathcal{P})^-] \vdash_R \bigwedge x\,.\,(G)^+$, which is just $[(\mathcal{P})^-] \vdash_R (\forall x.G)^+$.

$D \supset G$: $\mathcal{P} \vdash_O D \supset G$ iff $\mathcal{P}, D \vdash_O G$; by the hypothesis this is equivalent to having the sequent $[(\mathcal{P} \cup \{D\})^-] \vdash_R (G)^+$, which in turn is equivalent to having the sequent $[(\mathcal{P})^-] \vdash_R (\bigotimes (D)^-) \multimap (G)^+$, which is just $[(\mathcal{P})^-] \vdash_R (D \supset G)^+$.

Thus we arrive at the following theorem.

THEOREM 4.7 (PROVABILITY UNDER ENCODING) *Let $\mathcal{P}$ be a hereditary Harrop program and G be a hereditary Harrop goal. $\mathcal{P} \vdash G$ is provable in intuitionistic logic iff $[(\mathcal{P})^-] \vdash (G)^+$ is provable in linear logic.*

Finally, consider the linear encoding $(\mathcal{D})^\sim$ of $\mathcal{D}$ and the linear encoding $(G)^*$ of G given by:

$$
\begin{array}{lll}
(A)^* & =_{\mathrm{def}} & A \\
(G_1 \wedge G_2)^* & =_{\mathrm{def}} & (G_1)^* \,\&\, (G_2)^* \\
(G_1 \vee G_2)^* & =_{\mathrm{def}} & (G_1)^* \oplus (G_2)^* \\
(\exists \vec{x}.G)^* & =_{\mathrm{def}} & \bigvee \vec{x}\,.\,(G)^* \\
(\forall \vec{x}.G)^* & =_{\mathrm{def}} & \bigwedge \vec{x}\,.\,(G)^* \\
(D \supset G)^* & =_{\mathrm{def}} & (D)^\sim \multimap (G)^*
\end{array}
\qquad
\begin{array}{lll}
\mathcal{D}^\sim & =_{\mathrm{def}} & \bigcup_{D \in \mathcal{D}} (D)^\sim \\
(A)^\sim & =_{\mathrm{def}} & !\,A \\
(D_1 \wedge D_2)^\sim & =_{\mathrm{def}} & !\,(D_1)^\sim \otimes !\,(D_2)^\sim \\
(\forall \vec{x}.D)^\sim & =_{\mathrm{def}} & !\,(\bigwedge \vec{x}\,.\,(D)^\sim) \\
(G \supset A)^\sim & =_{\mathrm{def}} & !((G)^* \multimap A).
\end{array}
$$

We remark that this encoding is equivalent to that of Definition 4.5. Note that this latter encoding does not rely on the intuitionistic clausal decomposition $\|-\|$. The equivalence of the two given encodings relies on certain commutation properties of the clausal decompositions and those encodings.

5 Discussion

We have seen how, based on an analysis of the linear sequent calculus, a certain fragment of linear logic can be used as a linear logic programming language. This analysis consists of two, mutually dependent, main steps: (i) The identification of classes of definite formulae and goal formulae for which uniform proofs are complete (for logical consequence); (ii) The rewriting of programs to a clausal form for which clause-directed resolution proofs are complete (for logical consequence). It should be clear that there are several mutual dependencies amongst these steps, the precise elicitation of which remains the subject of active research, but crucial to which is the choice of which connectives can appear in definite formulae. Indeed, the question of whether the fragment considered here is a maximal one for which goal-directed, clause-directed resolution proof is available remains an open problem: it remains somewhat unclear what would constitute a precise technical statement of this property. We have also seen how first-order hereditary Harrop formulae can be used in our system via an encoding of this fragment of intuitionistic logic into linear logic, so that the system presented here constitutes a conservative extension of earlier work. In addition, it would seem that the linear extension has some interesting and novel features of its own, such as global variables. We will take up the theme of applications of the language elsewhere.

Whilst resolution proofs determine the logical nature of the search primitives needed to mechanically implement linear logic programming languages of the kind considered here, they are not, in operational terms, *deterministic*. For example, there is non-determinism such as that in the choice of a multiset of components that is to be used in the $\otimes$-rule and in the choice of resolvants. In order to describe *interpreters*, we must have a suitably deterministic operational characterization of resolution proofs. For example, consider the $\otimes$-rule (Definition 3.11):

$$\frac{\mathcal{D}_0 \vdash_R G_1, \mathcal{G}_0 \quad \mathcal{D}_0' \vdash_R G_2, \mathcal{G}_0' \quad \mathcal{D}_1 \vdash_R \mathcal{G}_1 \quad \ldots \quad \mathcal{D}_n \vdash_R \mathcal{G}_n}{\mathcal{D}, C \vdash_R \; G_1 \otimes G_2, \mathcal{G}},$$

where $\{\mathcal{D}_0, \mathcal{D}_0' \vdash_R \; G_1 \otimes G_2, \mathcal{G}_0, \mathcal{G}_0'\} \;\cup\; \bigcup_{i=1}^{n} \{\mathcal{D}_i \vdash_R \; \mathcal{G}_i\}$ is a multiset of C-components of the sequent $\mathcal{D}, C \vdash_R \; G_1 \otimes G_2, \mathcal{G}$. In this rule, read from conclusion to premisses, the program $\mathcal{D}$ and goal $\mathcal{G}$ are, roughly speaking, split, via C-components, into the parts $\mathcal{D}_0, \mathcal{D}_0', \mathcal{D}_1, \ldots, \mathcal{D}_n$ and $\mathcal{G}_0, \mathcal{G}_0', \mathcal{G}_1, \ldots, \mathcal{G}_n$, respectively. Recall that the following 2 is an example of this rule:

$$\frac{!(p \,\text{✠}\, q), p \vdash_R p \quad !(p \,\text{✠}\, q), p \vdash_R p \quad !(p \,\text{✠}\, q), q \vdash_R q \quad !(p \,\text{✠}\, q), q \vdash_R q}{!(p \,\text{✠}\, q) \vdash_R p, p, q \otimes q}.$$

The rules provides no determination of how the splitting, via C-components, is to be calculated. A similar problem arises with the resolution rule.

Our solution at the abstract level, discussed in [12] and in more detail in simpler settings in [10], [11], is to adopt a *lazy* approach. To this end we permit an interpreter to construct *proto-proofs*, by modifying the rules of resolution proof to delay the calculation of splittings; so that, for example, all suitable formulae in the antecedent and succedent go to each premiss at a $\otimes$-rule. In order to maintain soundness, we introduce the notion of *path* in such proto-proofs. Paths can be considered to be the proof-objects for proto-proofs — they are the appropriate notion of proof-object for our computational purposes and are related to proof-nets.

For a simple fragment of the logic of programs and goals, we sketch the definition of the construction of paths as follows: (I) The endsequent is in every path; (II) Traverse the proto-proof tree towards the leaves, starting at the endsequent; (i) Whenever a &-rule is reached, choose a branch and proceed; Whenever a $\otimes$- or resolution rule is reached, proceed along all branches; (III) Continue until all branches of the path have reached a leaf. The proto-proof determines a proof just in case for each possible path in it, there are expansions (at the appropriate rule applications) of the antecedent and succedent that are compatible with the leaves in the path.

We illustrate this technique by considering two examples (below). Suppose we are faced with the endsequent $!(p \otimes q) \vdash (q \multimap (q \otimes (p \otimes q))) \,\&\, (p \otimes q)$, which is provable in the linear sequent calculus. We must construct a resolution proof of $[!(p \otimes q)](=!(p \otimes q)) \vdash_R (q \multimap (q \otimes (p \otimes q))) \,\&\, (p \otimes q)$. Adopting our lazy approach, we obtain the following proto-proof:

$$\dfrac{\dfrac{\dfrac{\dfrac{!(p\otimes q), q \vdash^\sharp_R p \qquad !(p\otimes q), q \vdash^\sharp_R q}{!(p\otimes q), q \vdash^\sharp_R p \otimes q}\otimes \qquad !(p\otimes q), q \vdash^\sharp_R q}{!(p\otimes q), q \vdash^\sharp_R q \otimes (p \otimes q)}\otimes}{!(p\otimes q) \vdash^\sharp_R q \multimap (q \otimes (p \otimes q))}\multimap \qquad \dfrac{!(p\otimes q) \vdash^\flat_R p \qquad !(p\otimes q) \vdash^\flat_R q}{!(p\otimes q) \vdash^\flat_R p \otimes q}\otimes}{!(p\otimes q) \vdash^{\sharp\flat}_R (q \multimap (q \otimes (p \otimes q))) \,\&\, (p \otimes q)}\,\&\,.$$

Here there are two possible paths, one marked by $\sharp$ and one marked by $\flat$. The path marked by $\sharp$ is the more complicated. When calculating suitable splittings and expansions for this path, we must first use up the purely linear part of the antecedent, namely q, before considering the existence of a suitable expansion of $!(p \otimes q)$. Suppose that this q is used in the leftmost leaf, then it remains for us to check the existence of an expansion that is compatible with the remaining leaves, with q deleted from the antecedent, *i.e.* $!(p\otimes q) \vdash^\sharp_R p$ and $!(p \otimes q) \vdash^\sharp_R q$. Here we see that the expansion $\{ p, q \}$ is compatible with these leaves, so that we can conclude that this proto-proof does indeed determine a resolution proof of the given endsequent.

Our second example, based on (1), demonstrates that the notion of path can be extended to include definite formulae containing $\maltese$ and multiple-conclusioned sequents:

$$\dfrac{\dfrac{c, p, q_1, q_2 \vdash^\sharp_R p}{c, p, q_1, q_2, p \multimap r \vdash^\sharp_R r}\,\text{res.} \qquad \dfrac{c, p, q_1, q_2 \vdash^\sharp_R q_1 \qquad c, p, q_1, q_2 \vdash^\sharp_R q_2}{c, p, q_1, q_2 \vdash^\sharp_R q_1 \otimes q_2}\otimes}{p, q_1, q_2, !((p \multimap r) \maltese ((q_1 \otimes q_2) \multimap s)) \vdash^\sharp_R r, s}\,\text{res.}.$$

There is just one path, marked by $\sharp$, in this proto-proof. The two premisses of the first (lower) instance of the lazy resolution rule constitute a multiset of lazy $!((p \multimap r) \maltese ((q_1 \otimes q_2) \multimap s))$-components of the conclusion. The reader will easily verify that suitable expansions exist for a resolution proof to be determined by this proto-proof.

The theory of paths, which bears some similarity to the work of Andrews [2], Bibel [3], Wallen [22] on the use of matrices for proof-search in classical and modal logics, is the subject of current research (it is fully understood only for simple fragments, including the examples given above, of the logic of programs

and goals considered here) and will be reported elsewhere. Such a theory provides the transition from the logical characterization of goal-directed proof-search provided by resolution proofs to a truly operational account, whilst making no inessential commitment to a particular implementation strategy. We remark that, for a fragment of the logic of programs and goals considered here, the "input/output model of resource consumption", introduced in [13], can be used to implement the use of proto-proofs and paths.

We remark that it is important in the design of an interpreter to consider the permutability properties of right rules with respect to each other. The order of reduction of the formulae in the goal should respect the relative permutability of the right rules. We further remark that it is our judgement that it is inappropriate to consider the issue of a strategy for the selection of a goal from a multiset of goal formulae at the level of the definition of resolution proof: the computational appeal of resolution proofs lies in their treatment of left rules. The design of an interpreter will be discussed elsewhere.

Another question of interest is whether there is an interesting semantic analysis of the logic of programs and goals. In [10], an elementary construction of a suitable quantale of terms is given, by picking out the fragment of the term model [23] of full classical linear that is generated by the sequents of the logic of programs and goals. Whilst this construction yields a completeness theorem, it does not give an adequate account of the process of computation, such as that given by the least fixed point constructions of, for example, [16], [17]. The approach of Corradini and Montanari [4] promises to be valuable in our setting. Whilst the permutability analysis seems to indicate that uniform provability provides a basis for (linear) logic programming, it can be asked to what extent logic programming is characterized by this, or a similar, notion. A semantic analysis of the logic of definite formulae and goal formulae may help to answer this question, which is related to the question of maximality discussed above. Such an analysis will be given elsewhere.

6 Acknowledgements

The authors are grateful to Simon Ambler, Gianluigi Bellin, Jean-Yves Girard, Dale Miller, Gordon Plotkin and Lincoln Wallen for their encouragement of this work and to a referee for his suggestions.

Harland is supported by a grant of the Australian Research Council through the Machine Intelligence Project. Whilst at the University of Edinburgh, Pym was supported by U.K. Science and Engineering Research Council grant GR/G 58588, "Logical and Semantical Frameworks"; and by European Community ESPRIT Basic Research Action 3245, "Logical Frameworks: Design, Implementation and Experiment".

References

[1] J.-M. Andreoli and R. Pareschi. Linear Objects: Logical Processes with Built-in Inheritance. Proceedings of the International Conference on Logic Programming, pp. 496-510, Jerusalem, June, 1990. MIT Press, 1990.

[2] P.B. Andrews. Theorem-proving via general matings. *J. Assoc. Comp. Mach.* 28(2):193-214, 1981.

[3] W. Bibel. Computationally Improved Versions of Herbrand's Theorem. *Logic Colloquium '81*, pp. 11-28. North-Holland, 1982.

[4] A. Corradini and U. Montanari. An Algebraic Representation of Logic Program Computations. *Computational Logic*, Jean-Louis Lassez and Gordon Plotkin (editors), MIT Press, 1991, 584-612.

[5] H.B. Curry. The permutability of rules in the classical inferential calculus. *J. Symb. Log.* 17, pp. 245-248, 1952.

[6] G. Gentzen. Untersuchungen über das logische Schliessen. *Mathematische Zeitschrift* 39, pp. 176-210, 405-431, 1934.

[7] J.-Y. Girard. Linear Logic. *Theoret. Comp. Sci.* 50, pp. 1-102, 1987.

[8] J.-Y. Girard, Y. Lafont and P. Taylor. *Proofs and Types*. Cambridge University Press, 1989.

[9] J.A. Harland. A Proof-Theoretic Analysis of Goal-directed Provability. Accepted for publication in *J. Log. Comp.*. To appear.

[10] J.A. Harland and D.J. Pym. The Uniform Proof-theoretic Foundation of Linear Logic Programming. Report ECS-LFCS-90-124, University of Edinburgh, November 1990.

[11] J.A. Harland and D.J. Pym. The Uniform Proof-theoretic Foundation of Linear Logic Programming (Extended Abstract). Proceedings of the International Logic Programming Symposium, V. Saraswat and K. Ueda (editors), San Diego, October, 1991, 304-318, MIT Press, 1991.

[12] J.A. Harland and D.J. Pym. Resolution in Fragments of Classical Linear Logic (Extended Abstract). Proceedings of the Russian Conference on Logic Programming and Automated Reasoning (A. Voronkov, editor), St. Petersburg, July, 1992, *Lecture Notes in Artificial Intelligence*, Vol. 624, 30-41, Springer-Verlag, 1992.

[13] J. Hodas and D. Miller. Logic Programming in a Fragment of Intuitionistic Linear Logic: Extended Abstract. Proc. 6th Annual IEEE Symposium on Logic in Computer Science, Amsterdam, July 1991, IEEE Computer Society Press, 1991.

[14] S.C. Kleene. *Mathematical Logic.* Wiley and Sons, 1968.

[15] S.C. Kleene. Permutability of inferences in Gentzen's calculi LK and LJ. *Memoirs of the American Mathematical Society* **10**, pp. 1-26, 1952.

[16] J. Lloyd *Foundations of Logic Programming.* Springer-Verlag, 1984.

[17] D. Miller. A Logical Analysis of Modules in Logic Programming. *J. Log. Prog.* 6(1&2), pp. 79-108, 1989.

[18] D. Miller. Linear Logic and Logic Programming: An Overview. Proceedings of the JICSLP'92 Workshop on Linear Logic and Logic Programming, Washington DC, November, 1992. Published as Technical Report MS-CIS-92-80, Computer and Information Science Department, University of Philadelphia, 1992.

[19] D. Miller, G. Nadathur, F. Pfenning and A. Ščedrov. Uniform Proofs as a Foundation for Logic Programming. *Annals of Pure and Applied Logic* 51, pp. 125-157, 1991.

[20] D.J. Pym and L.A. Wallen. Proof-search in the $\lambda\Pi$-calculus. In: *Logical Frameworks*, G. Huet and G. Plotkin (editors), Cambridge University Press, 1991.

[21] D.J. Pym. Errata and Remarks. University of Edinburgh LFCS Report, ECS-LFCS-93-265, 1993.

[22] L.A. Wallen. *Automated Deduction in Non-classical Logics.* MIT Press, 1989.

[23] D. Yetter. Quantales and (Non-commutative) Linear Logic. *J. Symb. Log.* 55(1), pp. 41-64, 1990.

A Linear Sequent Calculus

$$\frac{}{\phi \vdash \phi}\ \text{axiom} \qquad \frac{\Gamma \vdash \phi, \Delta \qquad \Gamma', \phi \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}\ \text{cut}$$

$$\frac{\Gamma, \phi, \psi, \Gamma' \vdash \Delta}{\Gamma, \psi, \phi, \Gamma' \vdash \Delta}\ \text{X-L} \qquad \frac{\Gamma \vdash \Delta, \phi, \psi, \Delta'}{\Gamma \vdash \Delta, \psi, \phi, \Delta'}\ \text{X-R}$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, \mathbf{1} \vdash \Delta}\ \mathbf{1}\text{-L} \qquad \frac{}{\vdash \mathbf{1}}\ \mathbf{1}\text{-R}$$

$$\frac{}{\bot \vdash}\ \text{L}^{\bot} \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \bot, \Delta}\ \text{R}^{\bot}$$

$$\frac{}{\Gamma, \mathbf{0} \vdash \Delta}\ \mathbf{0}\text{-L} \qquad \frac{}{\Gamma \vdash \top, \Delta}\ \top\text{-R}$$

$$\frac{\Gamma \vdash \phi, \Delta}{\Gamma, \phi^{\perp} \vdash \Delta}\ \perp\text{-L} \qquad \frac{\Gamma, \phi \vdash \Delta}{\Gamma \vdash \phi^{\perp}, \Delta}\ \perp\text{-R}$$

$$\frac{\Gamma, \phi, \psi \vdash \Delta}{\Gamma, \phi \otimes \psi \vdash \Delta}\ \otimes\text{-L} \qquad \frac{\Gamma \vdash \phi, \Delta \qquad \Gamma' \vdash \psi, \Delta'}{\Gamma, \Gamma' \vdash \phi \otimes \psi, \Delta, \Delta'}\ \otimes\text{-R}$$

$$\frac{\Gamma, \phi \vdash \Delta}{\Gamma, \phi \,\&\, \psi \vdash \Delta} \qquad \frac{\Gamma, \psi \vdash \Delta}{\Gamma, \phi \,\&\, \psi \vdash \Delta}\ \&\text{-L} \qquad \frac{\Gamma \vdash \phi, \Delta \qquad \Gamma \vdash \psi, \Delta}{\Gamma \vdash \phi \,\&\, \psi, \Delta}\ \&\text{-R}$$

$$\frac{\Gamma, \phi \vdash \Delta \qquad \Gamma, \psi \vdash \Delta}{\Gamma, \phi \oplus \psi \vdash \Delta}\ \oplus\text{-L} \qquad \frac{\Gamma \vdash \phi, \Delta}{\Gamma \vdash \phi \oplus \psi, \Delta} \qquad \frac{\Gamma \vdash \psi, \Delta}{\Gamma \vdash \phi \oplus \psi, \Delta}\ \oplus\text{-R}$$

$$\frac{\Gamma, \phi \vdash \Delta \qquad \Gamma', \psi \vdash \Delta'}{\Gamma, \Gamma', \phi \between \psi \vdash \Delta, \Delta'}\ \between\text{-L} \qquad \frac{\Gamma \vdash \phi, \psi, \Delta}{\Gamma \vdash \phi \between \psi, \Delta}\ \between\text{-R}$$

$$\frac{\Gamma \vdash \phi, \Delta \qquad \Gamma', \psi \vdash \Delta'}{\Gamma, \Gamma', \phi \multimap \psi \vdash \Delta, \Delta'}\ \multimap\text{-L} \qquad \frac{\Gamma, \phi \vdash \psi, \Delta}{\Gamma \vdash \phi \multimap \psi, \Delta}\ \multimap\text{-R}$$

$$\frac{\Gamma, \phi \vdash \Delta}{\Gamma, !\phi \vdash \Delta}\ !\text{-L} \qquad \frac{!\Gamma \vdash \phi, ?\Delta}{!\Gamma \vdash !\phi, ?\Delta}\ !\text{-R}$$

$$\frac{!\Gamma, \phi \vdash ?\Delta}{!\Gamma, ?\phi \vdash ?\Delta}\ ?\text{-L} \qquad \frac{\Gamma \vdash \phi, \Delta}{\Gamma \vdash ?\phi, \Delta}\ ?\text{-R}$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, !\phi \vdash \Delta}\ W!\text{-L} \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash ?\phi, \Delta}\ W?\text{-R}$$

$$\frac{\Gamma, !\phi, !\phi \vdash \Delta}{\Gamma, !\phi \vdash \Delta}\ C!\text{-L} \qquad \frac{\Gamma \vdash ?\phi, ?\phi, \Delta}{\Gamma \vdash ?\phi, \Delta}\ C?\text{-R}$$

$$\frac{\Gamma, \phi[t/x] \vdash \Delta}{\Gamma, \bigwedge x.\phi \vdash \Delta}\ \bigwedge\text{-L} \qquad \frac{\Gamma \vdash \phi[y/x], \Delta}{\Gamma \vdash \bigwedge x.\phi, \Delta}\ \bigwedge\text{-R } (y \text{ not free in } \Gamma,\ \Delta)$$

$$\frac{\Gamma, \phi[y/x] \vdash \Delta}{\Gamma, \bigvee x.\phi \vdash \Delta}\ \bigvee\text{-L } (y \text{ not free in } \Gamma,\ \Delta) \qquad \frac{\Gamma \vdash \phi[t/x], \Delta}{\Gamma \vdash \bigvee x.\phi, \Delta}\ \bigvee\text{-R}$$