A system of inference based on proof search: an extended abstract

Dale Miller
Inria Saclay & LIX, Institut Polytechnique de Paris
Palaiseau France
Orcid: 0000-0003-0274-4954

***Abstract*—Gentzen designed his natural deduction proof system to "come as close as possible to actual reasoning." Indeed, natural deduction proofs closely resemble the static structure of logical reasoning in mathematical arguments. However, different features of inference are compelling to capture when one wants to support the process of searching for proofs. PSF (Proof Search Framework) attempts to capture these features naturally and directly. The design and metatheory of PSF are presented, and its ability to specify a range of proof systems for classical, intuitionistic, and linear logic is illustrated.**

***Index Terms*—proof systems, proof search, logical frameworks**

I. Introduction

Inference and proofs are often described using proof rules of various shapes. For example, natural deduction and sequent calculus use figures such as

$$\frac{\_ \quad \_ \quad \begin{matrix}(\_)\\ \vdots \\ \_\end{matrix}}{\_} \qquad \text{and} \qquad \frac{\_ \longrightarrow \_, \_ \qquad \_, \_ \longrightarrow \_}{\_, \_ \longrightarrow \_, \_}$$

These figures, introduced by Gentzen [1], rely on several punctuation marks such as the horizontal bar (to separate premises from conclusion), vertical dots (for reasoning from assumptions), parenthesized formulas (for discharging a formula), and the sequent arrow. The logical force implicit in the punctuation marks used to describe proofs can invade the logic specified in that framework. As Wittgenstein once stated: "Signs for logical operations are punctuation marks." (Tractatus 5.4611, 1922). While such influence of the framework might be hard to avoid in general, we should be aware of its influence and, at times, look for alternative systems of punctuation.

Gentzen declared that his natural deduction system NJ was "a formal system which comes as close as possible to actual reasoning" [1]. Indeed, his natural deduction proof systems have had great success ranging from being used in the teaching of logical reasoning to the formal encoding of proofs as dependently typed λ-terms. However, since Gentzen's introduction of such notation some four score and eight years ago, many different priorities for logic and proof have appeared.

While natural deduction and sequent calculus have been used successfully to describe the static structure of complete proofs (and their transformation via normalization and cut elimination), the dynamic structure of the *search* for proofs is less well captured by his systems. Here, issues such as *partial proofs*, *invertible inference rules*, and *don't care* and *don't know* non-determinism are particularly important to support.

II. Design motivated

Consider a sheet of paper on which a mathematician has written several formulas at the top and one at the bottom. Such a sheet is useful to represent a proof gap, where one needs to find a logical argument that connects the *given* formulas at the top to the intended *consequence* written at the bottom. In **PSF**, the search for a proof is encoded as the rewriting of a collection of such proof gaps recorded on sheets. A sheet might rewrite to no additional sheets if it is recognized as trivially proved: for example, because the formula at the bottom of the sheet is also present at the top. On the other hand, a sheet can rewrite to other sheets if solving those additional sheets is understood as a way to solving the originating sheet. For example, a sheet containing the formula $even(n) \vee odd(n)$ at the top can be rewritten to make two identical copies except that $even(n)$ is put at the top in one and $odd(n)$ is put at the top into the other. The rule of cases would justify such a rewriting. **PSF** encodes such sheets as *multisets* of *tagged* formulas: if the logical formula B appears at the top of the sheet, it is placed into that multiset as $\lfloor B \rfloor$; if it appears at the bottom, it is placed into that multiset as $\lceil B \rceil$ (see Section V-B).

A feature of inference rules that **PSF** puts in prominence is the difference between *multiplicative* and *additive* inference rules. The following are examples of the additive and multiplicative versions of the right introduction for conjunction.

$$\frac{\Gamma \vdash A, \Delta \quad \Gamma \vdash B, \Delta}{\Gamma \vdash A \wedge B, \Delta} \qquad \frac{\Gamma_1 \vdash A, \Delta_1 \quad \Gamma_2 \vdash B, \Delta_2}{\Gamma_1, \Gamma_2 \vdash A \wedge B, \Delta_1, \Delta_2}$$

More generally, an inference rule is *additive* if every side-formula occurrence (i.e., those in Γ and Δ) also occur in every premise. A rule is called *multiplicative* if every side-formula occurrence (i.e., those in Γ_1, Γ_2, Δ_1, and Δ_2) also occurs in exactly one premise. A rule with exactly one premise is necessarily both additive and multiplicative. **PSF** contains two operators $+$ and $\times$ responsible for injecting additive and multiplicative features into inference systems encoded into it.

The multiplicative features of **PSF** are easily illustrated by the need to rewrite multisets to other multisets. In particular, multisets will be encoded as expressions built from (some fixed set of) atomic expressions along with $\times$ for building a non-empty multiset and its unit $\mathbf{1}$ denoting an empty multiset. For example, if a, b, and c are atomic expressions, then $a \times a \times b$ denotes the multiset that contains two occurrences of a and one occurrence of b. Rewriting a multiset M to another multiset N using the rule $M_1 \mapsto M_2$ (where M_1 and M_2 are also multisets) is done using the following steps. (1) Split M into two parts M' and M''. (2) Determine that M' is the same multiset as M_1. (3) Identify N with the multiset union of M_2 and M''. The following small proof system (extended in the next section) can be used to describe such a computation.

$$\frac{\vdash \Delta}{\vdash \mathbf{1}, \Delta} \qquad \frac{\vdash E_1, E_2, \Delta}{\vdash E_1 \times E_2, \Delta} \qquad \frac{}{E \vdash E}$$

$$\frac{E_1 \vdash \Delta_1 \quad \vdash E_2, \Delta_2}{E_1 \mapsto E_2 \vdash \Delta_1, \Delta_2} \qquad \frac{}{\mathbf{1} \vdash} \qquad \frac{E_1 \vdash \Delta_1 \quad E_2 \vdash \Delta_2}{E_1 \times E_2 \vdash \Delta_1, \Delta_2}$$

The left-introduction rule for $\mapsto$ achieves the three steps mentioned above. Step (1) is captured by splitting a multiset into the union of Δ_1 and Δ_2 in that rule's conclusion. Steps (2) and (3) are captured by the proofs of its left and right premises, respectively. The rewriting of the multiset $\{a, a, b\}$ into $\{a, c\}$ by the rule that replaces a and b with c is witnessed by a derivation of $a \times b \mapsto c \vdash a, a, b$ from the open premise $\vdash a, c$.

Additive features are also incorporated into **PSF** using $+$ and its unit $\mathbf{0}$: in particular, collections of multisets are represented as a $+$ of $\times$ of atomic expressions. Below we list three additional features abstracted from searching for proofs based on evolving collections of sheets.

a) Linear and classical realms: When rewriting a sheet of paper to possibly other sheets, it is usually the case that some items are retained while others might disappear. In particular, an assumption at the top of a sheet is usually retained on all subproblems that are eventually rewritten from it, while the goal formula on one sheet may or may not change. For example, if the goal formula is $A \supset B$, then that goal is *replaced* by the goal formula B with A simultaneously added at the top of the sheet. The **PSF** recognizes this distinction by classifying atomic expressions as being in either the *linear realm*—where such expressions might be deleted or replaced—or the *classical realm*—where such expressions persist through all evolutions of a multiset. (There is a strong influence of linear logic [2] on the design of **PSF**.)

b) Bottom-up and top-down reasoning: These proof search styles appear in various different disguises in computational logic. They differentiate Prolog from Datalog and tableaux from resolution [3]. Term representation is often described using top-down proof structures, while term representations that allow for explicit sharing can be justified using bottom-up proof structures [4]. In **PSF**, this distinction comes into play using the notions of bias assignment and debts.

c) Don't care and don't know non-determinism: The non-determinism encountered in the search for proofs can be categorized as being either the *don't care* or *don't know* varieties. In **PSF**, inference rules will eventually be organized

THE RIGHT RULES

$$\frac{}{\Gamma \vdash \mathbf{0}, \Delta} \qquad \frac{\Gamma \vdash E_1, \Delta \quad \Gamma \vdash E_2, \Delta}{\Gamma \vdash E_1 + E_2, \Delta}$$

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \mathbf{1}, \Delta} \qquad \frac{\Gamma \vdash E_1, E_2, \Delta}{\Gamma \vdash E_1 \times E_2, \Delta}$$

THE LEFT RULES

$$\frac{}{\mathbf{1} \vdash} \qquad \frac{\Gamma, R_i \vdash \Delta}{\Gamma, R_1 + R_2 \vdash \Delta} \qquad \frac{\Gamma_1, R_1 \vdash \Delta_1 \quad \Gamma_2, R_2 \vdash \Delta_2}{\Gamma_1, \Gamma_2, R_1 \times R_2 \vdash \Delta_1, \Delta_2}$$

$$\frac{\Gamma_1, R \vdash \Delta_1 \quad \Gamma_2 \vdash E, \Delta_2}{\Gamma_1, \Gamma_2, R \mapsto E \vdash \Delta_1, \Delta_2} \qquad \frac{R \vdash \Upsilon \quad \Gamma \vdash E, \Delta, \Upsilon}{\Gamma, R \Mapsto E \vdash \Delta, \Upsilon}$$

$$\frac{\Gamma \vdash \Delta}{\vdash \Delta} \ \textit{decide},\ \Gamma \subseteq \mathcal{R} \text{ is non-empty and finite}$$

$$\frac{\Gamma \vdash \bar{A}, \Delta \quad \delta(A) = +1}{\Gamma, A \vdash \Delta} \ \textit{debit}_1$$

$$\frac{\vdash \bar{S}, \Upsilon \quad \delta(S) = +2}{S \vdash \Upsilon} \ \textit{debit}_2$$

THE IDENTITY RULES

$$\frac{}{E \vdash E} \ \textit{init} \qquad \frac{}{\vdash \bar{A}, A} \ \textit{iou}$$

THE STRUCTURAL RULES

$$\frac{\Gamma \vdash \Delta, S, S}{\Gamma \vdash \Delta, S} \ \textit{cR} \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, S} \ \textit{wR}$$

Fig. 1. The **B** proof system.

into two phases: the *right phase* will capture don't care non-determinism and the *left phase* will capture don't know non-determinism.

PSF is presented using two inference system. The basic system, **B**, is presented in Section III while a more structured variant, **F**, is presented in Section IV.

III. THE BASIC INFERENCE SYSTEM B

Fig. 1 contains the inference system **B**, which contains all the features we have motivated so far: additive and multiplicative structures, proof state rewriting, debts, and the linear and classical realms. The schematic variables used in Fig. 1 are the following. The variable A ranges over some fixed set of atomic expressions. The variables E and R range over expressions and rules and are defined as follows.

$$\begin{aligned} E &::= A \mid \mathbf{0} \mid E_1 + E_2 \mid \mathbf{1} \mid E_1 \times E_2 \\ R &::= A \mid \mathbf{0} \mid R_1 + R_2 \mid \mathbf{1} \mid R_1 \times R_2 \\ &\quad\ \mid R \mapsto E \mid R \Mapsto E \end{aligned}$$

The operators $\mapsto$ and $\Mapsto$ associate to the left while the operators $+$ and $\times$ associate to the right. A *debt* is an expressions of the form $\overline{A}$. The variable Γ ranges over multisets containing R-expressions, and the variable Δ ranges over multisets that can contain both E-expressions and debts. The variable $\mathcal{R}$ denotes some countable set of R-expressions. The function

$\delta(\cdot)$ is a *bias assignment*: it maps atomic expressions to the set $\{-2, -1, +1, +2\}$ (a similar bias assignment was used in [5]). The atomic expression A is in the *linear realm* if $\delta(A)$ is ± 1 and in the *classical realm* if $\delta(A)$ is ± 2. If $\delta(A) > 0$ then a debit rule can be used with A. The variable Υ ranges over finite multisets of atomic expressions in the classical realm, and the variable S ranges over atomic expressions in the classical realm.

A **B**-proof is *atomically closed* if all occurrences of the *init* rule in it involve only atomic expressions, i.e., they are of the form $A \vdash A$ for an atomic expression A.

Proposition 1 (Completeness of atomically closed **B***-proofs):* If the sequent $\vdash \Delta$ has a **B**-proof then it has an atomically closed **B**-proof.

Proof A simple induction on the structure of E shows that any occurrence of $E \vdash E$ in which E is not an atomic expression can be replaced by a proof that is atomically closed. □

The proofs below concerning the **B** proof system will implicitly apply the structural rules for atomic expressions with bias assignments of ± 2. In particular, the part of a context composed of just such atomic expressions, usually denoted with the Υ variable, will be treated additively even within multiplicative rules.

Proposition 2 (Clip-admissibility for **B***-proofs):* The following inference rule (a simpler version of Gentzen's cut rule) is admissible in **B**.

$$\frac{\Gamma_1 \vdash \Delta_1, E, \Upsilon \qquad \Gamma_2, E \vdash \Delta_2, \Upsilon}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Upsilon}\ \mathit{clip}$$

Proof Consider the following **B**-proof with exactly one occurrence of the *clip* rule.

$$\frac{\overset{\Xi_1}{\Gamma_1 \vdash \Delta_1, E, \Upsilon} \qquad \overset{\Xi_2}{\Gamma_2, E \vdash \Delta_2, \Upsilon}}{\Gamma_1, \Gamma_2 \vdash \Delta_1, \Delta_2, \Upsilon}\ \mathit{clip}$$

By Proposition 1, we can assume that both Ξ_1 and Ξ_2 are atomically closed. We proceed by considering the structures of Ξ_1 and Ξ_2. If either of these proofs ends in a right rule for Δ_1 or Δ_2, we can permute those rule occurrences down. Thus, we can assume that Δ_1 and Δ_2 are multisets of atomic expressions. Under this assumption, we can also permute down any left rule that might terminate Ξ_2. In this case, we can assume that Γ_2 is empty. All that is left is showing how to permute the *clip* rule up into the left premise proof.

Consider the following instance of *clip*. Here, E is either not atomic or it is atomic and $\delta(E) = \pm 1$.

$$\dfrac{\dfrac{\overset{\Xi_1}{R \vdash \Upsilon} \quad \overset{\Xi_2}{\Gamma_1 \vdash E, E', \Delta_1, \Upsilon}}{\Gamma_1, R \Mapsto E' \vdash E, \Delta_1, \Upsilon} \qquad \overset{\Xi_3}{E \vdash \Delta_2, \Upsilon}}{\Gamma_1, R \Mapsto E' \vdash \Delta_1, \Delta_2, \Upsilon}\ \mathit{clip}$$

This instant can be rewritten to be

$$\dfrac{\overset{\Xi_1}{R \vdash \Upsilon} \quad \dfrac{\overset{\Xi_2}{\Gamma_1 \vdash E, E', \Delta_1, \Upsilon} \quad \overset{\Xi_3}{E \vdash \Delta_2, \Upsilon}}{\Gamma_1 \vdash E', \Delta_1, \Delta_2, \Upsilon}\ \mathit{clip}}{\Gamma_1, R \Mapsto E' \vdash \Delta_1, \Delta_2, \Upsilon}$$

In the case that E is an atomic expression and $\delta(E) = \pm 2$ then the last inference rule of is either *init* (in which case, *clip* is easily removed since $E \in \Upsilon$) or debit_2 and, in that case, Δ_2 is empty (or a structural rule). In this final case, the proof above can be rewritten as

$$\dfrac{\dfrac{\overset{\Xi_1}{R \vdash E, \Upsilon} \quad \overset{\Xi_3}{E \vdash \Upsilon}}{R \vdash \Upsilon}\mathit{clip} \quad \dfrac{\overset{\Xi_2}{\Gamma_1 \vdash E, E', \Delta_1, \Upsilon} \quad \overset{\Xi_3}{E \vdash \Upsilon}}{\Gamma_1 \vdash E', \Delta_1, \Upsilon}\mathit{clip}.}{R \Mapsto E' \vdash \Delta_1, \Upsilon}$$

The other cases regarding the structure of the R-expression in Γ are simple and direct.

The only remaining cases to consider is when Ξ_1 is a right rule introducing E and Ξ_2 is a left rule introducing E. These cases are discussed below (remembering that Γ_2 is empty).

It is not possible for E to be $\mathbf{0}$ since there is no such proof Ξ_2. If E is $\mathbf{1}$, then Δ_2 is empty and Ξ_1 replaces the clip rule. If E is $E_1 + E_2$ then we must have

$$\dfrac{\dfrac{\overset{\Xi_1'}{\Gamma_1 \vdash \Delta_1, E_1} \quad \overset{\Xi_1''}{\Gamma_1 \vdash \Delta_1, E_2}}{\Gamma_1 \vdash \Delta_1, E_1 + E_2} \qquad \dfrac{\overset{\Xi_2'}{E_1 \vdash \Delta_2}}{E_1 + E_2 \vdash \Delta_2}}{\Gamma_1 \vdash \Delta_1, \Delta_2}\ \mathit{clip}$$

(where Δ_1 and Δ_2 contain only atomic expressions). This instance of *clip* can be replaced by the following instance of *clip* on smaller expressions..

$$\dfrac{\overset{\Xi_1'}{\Gamma_1 \vdash \Delta_1, E_1} \quad \overset{\Xi_2'}{E_1 \vdash \Delta_2}}{\Gamma_1 \vdash \Delta_1, \Delta_2}\ \mathit{clip}$$

The symmetric case is handled the same. If E is $E_1 \times E_2$ then we must have

$$\dfrac{\dfrac{\overset{\Xi_1'}{\Gamma_1 \vdash \Delta_1, E_1, E_2}}{\Gamma_1 \vdash \Delta_1, E_1 \times E_2} \quad \dfrac{\overset{\Xi_2'}{E_1 \vdash \Delta_2'} \quad \overset{\Xi_2''}{E_2 \vdash \Delta_2''}}{E_1 \times E_2 \vdash \Delta', \Delta_2''}}{\Gamma_1 \vdash \Delta_1, \Delta_2}\ \mathit{clip}$$

(where Δ_1 and Δ_2 contain only atomic expressions). This instance of *clip* can be replaced by the following instance of *clip* on smaller expressions.

$$\dfrac{\dfrac{\overset{\Xi_1'}{\Gamma_1 \vdash \Delta_1, E_1, E_2} \quad \overset{\Xi_2'}{E_1 \vdash \Delta_2'}}{\Gamma_1 \vdash \Delta_1, \Delta_2', E_2}\ \mathit{clip} \quad \overset{\Xi_2''}{E_2 \vdash \Delta_2''}}{\Gamma_1 \vdash \Delta_1, \Delta_2', \Delta_2'}\ \mathit{clip}$$

In general, one occurrence of clip can be replaced by two clips. Standard induction arguments can now be used to complete this proof. □

Proposition 3 (Right rules are invertible): The right rules are invertible. In particular, if E is not atomic and the sequent $\vdash E, \Delta$ is provable, then there is a proof of this sequent in which the last inference rule is an introduction rule for E.

Proof Let Ξ be a proof of $\vdash E_1 \times E_2, \Delta$. Consider

$$\dfrac{\dfrac{\stackrel{\Xi}{\vdash E_1 \times E_2, \Delta} \qquad \dfrac{\dfrac{}{E_1 \vdash E_1}\,\mathit{init} \quad \dfrac{}{E_2 \vdash E_2}\,\mathit{init}}{E_1 \times E_2 \vdash E_1, E_2}}{\Gamma \vdash E_1, E_2, \Delta}\,\mathit{clip}}{\Gamma \vdash E_1 \times E_2, \Delta}$$

By Proposition 2, this proof with *clip* can be replaced by a proof without *clip*: that proof ends in the introduction of $E_1 \times E_2$. The case where E is **1** is similar and simpler. Let Ξ be a proof of $\vdash E_1 + E_2, \Delta$. Consider

$$\dfrac{\dfrac{\stackrel{\Xi}{\vdash E_1 + E_2, \Delta} \qquad \dfrac{\dfrac{}{E_1 \vdash E_1}\,\mathit{init}}{E_1 + E_2 \vdash E_1}}{\Gamma \vdash \Delta, E_1}\,\mathit{clip} \qquad \dfrac{\text{similar}}{\Gamma \vdash \Delta, E_2}\,\mathit{clip}}{\Gamma \vdash \Delta, E_1 + E_2}$$

By Proposition 2, this proof with *clip* can be replaced by a proof without *clip*. The case where E is **0** is immediate. □

Proposition 4 (Clipping out debt): Let C be an atomic expression. If $\delta(C) = +1$, the following dclip_1 rule is admissible.

$$\dfrac{\vdash \Delta_1, C \qquad \vdash \Delta_2, \overline{C}}{\vdash \Delta_1, \Delta_2}\,\mathit{dclip}_1$$

If $\delta(C) = +2$, the following dclip_2 rule is admissible.

$$\dfrac{\vdash \Delta, C \qquad \vdash \Upsilon, \overline{C}}{\vdash \Delta, \Upsilon}\,\mathit{dclip}_2$$

Proof Consider the following **B**-proof with one occurrence of dclip_1: here, $\delta(C) = +1$.

$$\dfrac{\stackrel{\Xi_1}{\vdash \Delta_1, C} \qquad \stackrel{\Xi_2}{\vdash \Delta_2, \overline{C}}}{\vdash \Delta_1, \Delta_2}\,\mathit{dclip}_1$$

This proof can be rewritten as

$$\dfrac{\stackrel{\Xi_1}{\vdash \Delta_1, C} \qquad \dfrac{\stackrel{\Xi_2}{\vdash \Delta_2, \overline{C}}}{C \vdash \Delta_2}\,\mathit{debit}_1}{\vdash \Delta_1, \Delta_2}\,\mathit{clip}$$

Apply Proposition 2 to finish this case. Consider the following proof with one occurrence of dclip_2: here, $\delta(C) = +2$.

$$\dfrac{\stackrel{\Xi_1}{\vdash \Delta, C} \qquad \stackrel{\Xi_2}{\vdash \Upsilon, \overline{C}}}{\vdash \Delta, \Upsilon}\,\mathit{dclip}_2$$

This proof can be rewritten as

$$\dfrac{\stackrel{\Xi_1}{\vdash \Delta, C} \qquad \dfrac{\stackrel{\Xi_2}{\vdash \Upsilon, \overline{C}}}{C \vdash \Upsilon}\,\mathit{debit}_2}{\vdash \Delta, \Upsilon}\,\mathit{clip}$$

Apply Proposition 2 to finish this case. □

If an atomic expression A has a positive bias value, the *debit* rule allows turning an obligation into find A in the current multiset into a promise to pay that obligation later, possibly after additional rewriting takes place. The proposition theorem states that once a complete proof is built, possibly using the *debit* rules, it is possible to reorganize that proof so that no *debit* rules are used.

Proposition 5 (Completeness without debit): If the sequent $\vdash \Delta$ has a **B**-proof, it has a proof without the debit_1 and debit_2 rules.

Proof We systematically replace an occurrence of the debit_1 inference rule (above a *decide* rule) with *init* and dclip_1 (below the *decide* rule). That is, we transform

$$\dfrac{\dfrac{\cdots \qquad \begin{array}{c}\dfrac{\stackrel{\Xi}{\vdash \overline{A}, \Delta_0}}{A \vdash \Delta_0}\,\mathit{debit}_1 \\ \vdots\end{array} \qquad \cdots}{\Gamma \vdash \Delta_0, \Delta_1}}{\vdash \Delta_0, \Delta_1}\,\mathit{decide}$$

into the following proof containing dclip_1. Here, we replaced Δ_0 with A in some of the sequents and then used the *clip* rule to reintroduce the Δ_0 expressions.

$$\dfrac{\dfrac{\dfrac{\cdots \qquad \begin{array}{c}\dfrac{}{A \vdash A}\,\mathit{init} \\ \vdots\end{array} \qquad \cdots}{\Gamma \vdash A, \Delta_1}}{\vdash A, \Delta_1}\,\mathit{decide} \qquad \stackrel{\Xi}{\vdash \overline{A}, \Delta_0}}{\vdash \Delta_0, \Delta_1}\,\mathit{dclip}_1$$

We also can systematically replace an occurrence of the debit_2 inference rule (above a *decide* rule) with *init* and dclip_2 (below the *decide* rule). That is, we transform a **B**-proof of the form

$$\dfrac{\dfrac{\cdots \qquad \begin{array}{c}\dfrac{\stackrel{\Xi}{\vdash \overline{A}, \Upsilon}}{A \vdash \Upsilon}\,\mathit{debit}_1 \\ \vdots\end{array} \qquad \cdots}{\Gamma \vdash \Upsilon, \Delta}}{\vdash \Upsilon, \Delta}\,\mathit{decide}$$

with the following proof with dclip_2 below. Here, we replaced Υ with A in some of the sequents and used the dclip_2 rule to reintroduce the Υ expressions.

$$\dfrac{\dfrac{\dfrac{\cdots \qquad \begin{array}{c}\dfrac{}{A \vdash A}\,\mathit{init} \\ \vdots\end{array} \qquad \cdots}{\Gamma \vdash A, \Delta}}{\vdash A, \Delta}\,\mathit{decide} \qquad \stackrel{\Xi}{\vdash \overline{A}, \Upsilon}}{\vdash \Upsilon, \Delta}\,\mathit{dclip}_2$$

Thus, we have replaced one occurrence of either debit_1 or debit_2 with one occurrence of dclip_1 or dclip_2, respectively. Using Proposition 4, we have a clip-free proof with one fewer debit rules. Note that clip elimination does not introduce *debit* when there is no *debit* in the original proof. □

A **B**-proof Ξ is *reduced* if every occurrence of the *decide* rule has a right-hand side containing only atomic expressions or debts.

The *major premises* of the left rules are defined as follows. Those rules with only a single premise have that sole premise as their major premise. Both premises of the left-introduction rule for $\times$ are major premises. Finally, the left-most premise is the major premise for the introduction rules for $\mapsto$ and $\Mapsto$. Note that if the right-hand side of the conclusion of a left rule occurrence contains only atomic expressions, then this is true of the major premises of that rule occurrence.

Proposition 6 (Completeness of reduced **B***-proofs):* If the sequent $\vdash \Delta$ has a **B**-proof, it has a reduced proof.

Proof An occurrence of a sequent in Ξ is *bad* if that sequent is the conclusion of a left rule and a major premise of that rule is the conclusion of a right-introduction rule. Note that the right-hand side of a bad sequent occurrence must contain a non-atomic expression. The measure of a bad occurrence of a sequent is the height of its subproof in Ξ. The measure of the **B**-proof Ξ is the multiset of the measure of all bad sequents in Ξ. We prove that if the measure of Ξ is not the empty multiset, then we can replace Ξ with another proof of the same end-sequent but with strictly smaller multiset ordering.

Assume that the measure of Ξ is non-empty. Then there exists a sequent with a bad occurrence in Ξ. Pick one of these with minimal height and assume that that sequent is of the form $\Gamma \vdash \Delta$. As noted above, there must be a non-atomic expression in Δ. Hence, the last left rule cannot be either $debit_2$ or the left-introduction rule for **1**. Thus, we only need to consider six left rules (*decide*, $debit_1$, and one each for $\times, +, \mapsto, \Mapsto$). Since there are four right introduction rules (one for each of $\mathbf{1}, \times, \mathbf{0}, +$) then we have 24 possible combinations of rules that can yield the bad occurrence $\Gamma \vdash \Delta$. If the upper rule is the right-introduction of **0** or **1**, then we can trivially permute that rule down. We illustrate a few more cases. The remaining ones are similar.

$$\dfrac{\dfrac{\overset{\Xi_1}{R_1 \vdash E_1, \Delta_1} \quad \overset{\Xi_2}{R_1 \vdash E_2, \Delta_1}}{R_1 \vdash E_1 + E_2, \Delta_1} \quad \overset{\Xi_3}{R_2 \vdash \Delta_3}}{R_1 \times R_2 \vdash E_1 + E_2, \Delta_1, \Delta_2} \longrightarrow$$

$$\dfrac{\dfrac{\overset{\Xi_1}{R_1 \vdash E_1, \Delta_1} \quad \overset{\Xi_3}{R_2 \vdash \Delta_2}}{R_1 \times R_2 \vdash E_1, \Delta_1, \Delta_2} \quad \dfrac{\overset{\Xi_2}{R_1 \vdash E_2, \Delta_1} \quad \overset{\Xi_3}{R_2 \vdash \Delta_2}}{R_1 \times R_2 \vdash E_2, \Delta_1, \Delta_2}}{R_1 \times R_2 \vdash E_1 + E_2, \Delta_1, \Delta_2}$$

$$\dfrac{\dfrac{\overset{\Xi_1}{R \vdash E_1, E_2, \Delta_1}}{R \vdash E_1 \times E_2, \Delta_1} \quad \overset{\Xi_2}{\vdash E, \Delta_2}}{R \mapsto E \vdash E_1 \times E_2, \Delta_1, \Delta_2} \longrightarrow$$

$$\dfrac{\dfrac{\overset{\Xi_1}{R \vdash E_1, E_2, \Delta_1} \quad \overset{\Xi_2}{\vdash E, \Delta_2}}{R \mapsto E \vdash E_1, E_2, \Delta_1, \Delta_2}}{R \mapsto E \vdash E_1 \times E_2, \Delta_1, \Delta_2}$$

(We have assumed that these sequents have a left-hand side with at most two expressions: these cases are easily extended to the more general case.) Note that in the last pair of proofs, for example, the bad occurrence of the sequent is moved up, but the sequent $R \mapsto E \vdash E_1, E_2, \Delta_1$ may be a bad occurrence in the result: if that is the case, its measure has decreased. In this way, the measure decreases whenever we permute such rules. □

Since the *decide* rule in **B** allows for deciding on Γ with multiple expressions, we say that **B**-proofs are, in general, *multi-decide* proofs. A **B**-proof is a *single-decide* proof if every occurrence of the *decide* rule in it decides on exactly one expression. While allowing multi-decide proofs was a convenience for proving the clip-elimination result (Proposition 2), we maintain completeness by restricting to single-decide proofs.

Proposition 7 (Completeness of single-decide proofs): A **B** provable sequent has single-decide **B**-proof.

Proof In principle, deciding on multiple expressions can be done sequentially. Since all left rules permute over each other, we can assume that the left rules are done in a *focused* manner: that is, the immediate subexpressions of an R-expressions in major premises can be introduced in the proof of that major premise (we also include the use of *init* or a *debit* rule). Schematically, we can then take instances of the *decide* rule of the form

$$\dfrac{\dfrac{\cdots \quad \begin{array}{c} \underline{\Gamma_i \vdash \Delta_i} \\ \vdots \end{array} \quad \cdots}{R, \Gamma \vdash \Delta}}{\vdash \Delta}\ \mathit{decide}$$

where Γ is non-empty and Γ_i is a sub-multiset of Γ and where $i \in \{1, \ldots, n\}$ for some positive n. If Γ_i is non-empty, then we can transform this proof into

$$\dfrac{\dfrac{\begin{array}{c} \cdots \\ \vdots \\ \cdots \end{array} \quad \begin{array}{c} \dfrac{\Gamma_i \vdash \Delta_i}{\vdash \Delta_i}\ \mathit{decide} \\ \overline{} \\ \vdots \end{array} \quad \begin{array}{c} \cdots \\ \vdots \\ \cdots \end{array}}{R \vdash \Delta}}{\vdash \Delta}\ \mathit{decide}$$

An inductive argument can be used to remove all *decide* rules that decide on more than one rule. □

IV. The two-phase inference system F

The **B** inference system supports the basic features we motivated in Section II that should be present in an inference system that supports the search for proofs. At the same time, **B** can be improved significantly to better support such search.

In the previous section, we have taken steps in that direction already. The completeness of single-decide proofs means that we do not need to consider selecting collections of rules at a time because selecting them one at a time is just as complete. Similarly, the completeness of reduced proofs implies that the search for proofs can be done by first doing all possible right rules, then selecting one R-expression for the *decide* rule, and then doing only left rules along the major premises.

$$\frac{}{\vdash \mathbf{0},\Delta} \qquad \frac{\vdash E_1,\Delta \quad \vdash E_2,\Delta}{\vdash E_1+E_2,\Delta}$$

$$\frac{\vdash \Delta}{\vdash \mathbf{1},\Delta} \qquad \frac{\vdash E_1,E_2,\Delta}{\vdash E_1\times E_2,\Delta}$$

$$\frac{\Downarrow R\vdash \mathcal{A},\Upsilon}{\vdash \mathcal{A},\Upsilon}\ \textit{decide},\ R\in\mathcal{R} \qquad \frac{\Downarrow R_i\vdash \mathcal{A},\Upsilon}{\Downarrow R_1+R_2\vdash \mathcal{A},\Upsilon}$$

$$\frac{}{\Downarrow \mathbf{1}\vdash \Upsilon} \qquad \frac{\Downarrow R_1\vdash \mathcal{A}_1,\Upsilon \quad \Downarrow R_2\vdash \mathcal{A}_2,\Upsilon}{\Downarrow R_1\times R_2\vdash \mathcal{A}_1,\mathcal{A}_2,\Upsilon}$$

$$\frac{\Downarrow R\vdash \mathcal{A}_1,\Upsilon \quad \vdash E\Downarrow \mathcal{A}_2,\Upsilon}{\Downarrow R\mapsto E\vdash \mathcal{A}_1,\mathcal{A}_2,\Upsilon} \qquad \frac{\Downarrow R\vdash \mathcal{A} \quad \vdash E\Downarrow \Delta,\mathcal{A}}{\Downarrow R\Mapsto E\vdash \Delta,\mathcal{A}}$$

$$\frac{\vdash E,\mathcal{A},\Upsilon}{\vdash E\Downarrow \mathcal{A},\Upsilon}\ \textit{release}\dagger \qquad \frac{\delta(A)>0}{\vdash A\Downarrow \overline{A},\Upsilon}\ \textit{initR}$$

$$\frac{\delta(A)<0}{\Downarrow A\vdash A,\Upsilon}\ \textit{initL} \qquad \frac{\delta(A)>0}{\vdash \overline{A},A,\Upsilon}\ \textit{iou}$$

$$\frac{\vdash \overline{A},\mathcal{A},\Upsilon \quad \delta(A)=+1}{\Downarrow A\vdash \mathcal{A},\Upsilon}\ \textit{debit}_1$$

$$\frac{\vdash \bar{A},\Upsilon \quad \delta(A)=+2}{\Downarrow A\vdash \Upsilon}\ \textit{debit}_2$$

Fig. 2. The two-phase inference system **F**. The proviso † for the *release* rule states that E is either not atomic or it is atomic and $\delta(E)<0$.

There are, however, still defects in the search for proofs since there remains some non-determinism in the search for (reduced and single-decide) **B**-proofs that can be removed. For example, $A\vdash A$ can be proved using *init*, but, if $\delta(A)=+1$, it can also be proved using both $\textit{debit}_1$ and *iou*. Also, the rules of contraction and weakening can be applied at almost any moment during search.

The two-phased proof system in **F**, given in Fig. 2, captures only reduced and single-decide proofs and where these two non-deterministic choices are resolved. There are two kinds of sequents in **F**, namely $\vdash \Delta$ and $\Downarrow R\vdash \mathcal{A}$, where Δ is a multiset of E-expressions, R is an R-expression, and $\mathcal{A}$ is a multiset of atomic expressions. When comparing this proof system to **B**, there is a clear separation on left and right rules. A sequent of the form $\vdash \mathcal{A}$ is called a *border* sequent.

Note that in **F**, if $\delta(A)=+1$ and we encounter $\Downarrow A\vdash A$, then only the $\textit{debit}_1$ and *iou* rules can be used to prove it: the *initL* rule is not available. Also, the two structural rules are built into this proof system using the schematic variable Υ to denote a multiset of atomic expressions in the classical realm: this is achieved by treating the part of the context identified as Υ as additive even in multiplicative rules.

The proof of the following relative completeness theorem for **F** proofs follows from the completeness for reduced and single-decide proofs (Propositions 6 and 7).

Proposition 8: Let Δ be a multiset containing E-expressions and debits. Then, $\vdash \Delta$ is provable in **B** if and only if $\vdash\Delta$ is provable in **F**.

The proof system **F** is a *two-phase* proof system since all of its inference rules can be organized into the following two phases. A *left phase* is a derivation composed of only left rules and $\Downarrow$ sequents: this phase has a border sequent as its conclusion, and its premises are the conclusion of either *release*, $\textit{debit}_1$, or $\textit{debit}_2$. There are possibly many choices to make during the construction of a left phase (the choice of $R\in\mathcal{R}$, the choice of i in the left rule for $+$, and the choice of how to split the side expressions among premises) and, as a result, this phase encapsulates *don't know non-determinism*. A *right phase* is a derivation composed of only right rules: all of the premises of this phase are border sequents, and its conclusion is either the conclusion of the full proof or is the premise of either *release*, $\textit{debit}_1$, or $\textit{debit}_2$. Note that there might be many ways to build a right phase formally but they all relate their conclusion to the same collection of premises. In this sense, right phases encapsulate *don't care non-determinism*.

A *synthetic rule* is composed of one left phase and zero or more right phases, one for each premise of the left phase. In particular, the conclusion and all the premises of a synthetic rule are border sequents. We say that a synthetic rule is *for* R if the last rule (necessarily a *decide* rule) decides on R.

Note that the right rules and, hence, the right phase seen as a single rule, is additive (see Section II). If there are no atomic expressions with bias assignment ± 2 then the left rules, and the left phase seen as a single rule, are multiplicative. If atomic expressions have bias ± 2 then these are treated additively even in otherwise multiplicative rules.

The primary purpose of the **F** proof system over the **B** proof system is that the former is used to generate synthetic inference rules from R expressions. In the next section, we provide several illustrations of how R expressions can be used to specify various proof systems involving logical formulas.

V. Applications of B and F

A. Specifying Fibonacci numbers

Denote by $f(n)$ the n^{th} Fibonacci number and let $\mathcal{R}$ be the union of $\{F(0,0),\ F(1,1)\}$ and the set

$$\{F(n+2,x+y)\mapsto F(n+1,x)\mapsto F(n,y)\mid n,x,y\in Z\}.$$

To determine the synthetic rules that can arise from $\mathcal{R}$, consider the three cases for the value of $\delta(F(\cdot,\cdot))$.

If $\delta(F(\cdot,\cdot))<0$ then the synthetic rules are

$$\frac{}{\vdash F(0,0)} \qquad \frac{}{\vdash F(1,1)} \qquad \frac{\vdash F(n+1,x) \quad \vdash F(n,y)}{\vdash F(n+2,x+y)}$$

The sequent $\vdash F(n,f(n))$ is has a unique proof using these rules, and its size is exponential in n.

If $\delta(F(\cdot,\cdot))=+1$: then the synthetic rules are

$$\frac{\vdash \Delta,\overline{F(0,0)}}{\vdash \Delta} \qquad \frac{\vdash \Delta,\overline{F(1,1)}}{\vdash \Delta} \qquad \frac{\vdash \overline{F(n+2,x+y)},\Delta}{\vdash \overline{F(n+1,x)},\overline{F(n,y)},\Delta}$$

The sequent $\vdash F(n,f(n))$ is provable and the sizes of such proofs are exponential in n. While bottom-up reasoning is taking place, contraction is not available on debts. As a result, there is no sharing of previous computations.

Finally, if $\delta(F(\cdot,\cdot)) = +2$, then the synthetic rules are the same as the previous case except that Δ must be replaced with Υ.

$$\frac{\vdash \Upsilon, \overline{F(0,0)}}{\vdash \Upsilon} \quad \frac{\vdash \Upsilon, \overline{F(1,1)}}{\vdash \Upsilon} \quad \frac{\vdash \overline{F(n+2, x+y)}, \Upsilon}{\vdash \overline{F(n+1,x)}, \overline{F(n,y)}, \Upsilon}$$

The sequent $\vdash F(n, f(n))$ is provable only when $n \leq 3$.

Another specification of the Fibonacci series uses a more deliberate reuse strategy. Let $n \geq 0$ and let $\mathcal{R}_n$ be the set of rules that is the union of the singleton $\{F(n, f(n)) \mapsto \mathbf{0}\}$ and all the rules of the form

$$F(m+1, x) \times F(m, y) \mapsto F(m+2, x+y) \times F(m+1, x)$$

where m, x, y are natural numbers. In this case, the sequent $\vdash F(0,0) \times F(1,1)$ is provable from $\mathcal{R}_n$ with a proof of size linear in n.

B. *Classical and intuitionistic logic*

The main reason to introduce **PSF**, via the **B** and **F** proof systems, is to provide a specification framework for inference rules. When comparing different proof systems (e.g., a target and an encoding of it), three *levels of adequacy* naturally arise [6]. The weakest level of adequacy is *relative completeness*, which considers only *provability*: a formula has a proof in one system if it has a proof in the other system. A stronger level of adequacy is of *full completeness of proofs*: the proofs in one system are in one-to-one correspondence with proofs in the other system. If one uses the term "derivation" for possibly incomplete proofs (proofs that may have open premises), then the strongest version of adequacy is that of *full completeness of derivations*, where every derivation (such as inference rules themselves) are in one-to-one correspondence with those in the other system.

Unless otherwise mentioned, the encodings of proof systems described below will all be at the highest level of adequacy. In particular, one inference rule in a target proof system (say, a rule in natural deduction) will correspond to a synthetic rule in **F**.

Recalling now the discussion in Section II regarding representing the state of the search for a proof as a collection of sheets, these sheets are represented as multisets of atomic expressions of the form $\lfloor B \rfloor$ and $\lceil B \rceil$, where B denotes a logical formula. Here, the expression $\lfloor B \rfloor$ tags B as coming at the top of a sheet while $\lceil B \rceil$ tags B as coming at the bottom of a sheet.

The rules in Fig. 3 can be used to describe natural deduction in intuitionistic logic *and* the sequent calculus for both intuitionistic and classical logic. In all of these cases, classical logic is captured using the polarities $\delta(\lfloor \cdot \rfloor) = \pm 2$ and $\delta(\lceil \cdot \rceil) = \pm 2$. In contrast, intuitionistic logic is captured using the polarities $\delta(\lfloor \cdot \rfloor) = \pm 2$ and $\delta(\lceil \cdot \rceil) = \pm 1$. Here, we are considering only propositional logic with the logical constants $\supset$ (implication), $\wedge$ (conjunction), $\vee$ (disjunction), $\top$ (truth), and $\bot$ (false). (First-order quantification is addressed in Section V-H.)

$$\begin{array}{lrl}
(\supset L^m) & \lfloor A \supset B \rfloor \mapsto & \lceil A \rceil \Rrightarrow \lfloor B \rfloor \\
(\supset R^m) & \lceil A \supset B \rceil \mapsto & \lfloor A \rfloor \times \lceil B \rceil \\
(\wedge L^a) & \lfloor A \wedge B \rfloor \mapsto & \lfloor A \rfloor \\
(\wedge R^a) & \lceil A \wedge B \rceil \mapsto & \lceil A \rceil + \lceil B \rceil \\
(\wedge L^a) & \lfloor A \wedge B \rfloor \mapsto & \lfloor B \rfloor \\
(\vee L^a) & \lfloor A \vee B \rfloor \mapsto & \lfloor A \rfloor + \lfloor B \rfloor \\
(\vee R^a) & \lceil A \vee B \rceil \mapsto & \lceil A \rceil \\
(\vee R^a) & \lceil A \vee B \rceil \mapsto & \lceil B \rceil \\
(\bot L^a) & \lfloor \bot \rfloor \mapsto & \mathbf{0} \\
(\top R^a) & \lceil \top \rceil \mapsto & \mathbf{0} \\
(Id_1) & \lfloor C \rfloor \times \lceil C \rceil & \\
(Id_2) & \mathbf{1} \mapsto & \lceil C \rceil \Rrightarrow \lfloor C \rfloor
\end{array}$$

Fig. 3. Rules used to specify classical and intuitionistic logic. The superscript a and m on the names associated to R-expressions identify that rule as either additive or multiplicative.

$$\frac{\Gamma \vdash A \supset B \downarrow \quad \Gamma \vdash A \uparrow}{\Gamma \vdash B \downarrow} [\supset E] \qquad \frac{\Gamma, A \vdash B \uparrow}{\Gamma \vdash A \supset B \uparrow} [\supset I]$$

$$\frac{\Gamma \vdash A \wedge B \downarrow}{\Gamma \vdash A \downarrow} [\wedge E] \qquad \frac{\Gamma \vdash A \wedge B \downarrow}{\Gamma \vdash B \downarrow} [\wedge E]$$

$$\frac{\Gamma \vdash A \uparrow \quad \Gamma \vdash B \uparrow}{\Gamma \vdash A \wedge B \uparrow} [\wedge I]$$

$$\frac{}{\Gamma, A \vdash A \downarrow} [\mathrm{I}] \qquad \frac{\Gamma \vdash A \downarrow}{\Gamma \vdash A \uparrow} [\mathrm{M}] \qquad \frac{\Gamma \vdash A \uparrow}{\Gamma \vdash A \downarrow} [\mathrm{S}]$$

$$\frac{}{\Gamma \vdash \top \uparrow} [\top I] \qquad \frac{\Gamma \vdash \bot \downarrow}{\Gamma \vdash C \uparrow} [\bot E]$$

Fig. 4. The rules for the $\supset$, $\forall_i$, and $\wedge$ fragment of intuitionistic natural deduction NJ.

$$\frac{\Gamma \vdash A \vee B \downarrow \quad \Gamma, A \vdash C \uparrow(\downarrow) \quad \Gamma, B \vdash C \uparrow(\downarrow)}{\Gamma \vdash C \uparrow(\downarrow)} [\vee E]$$

$$\frac{\Gamma \vdash A_i \uparrow}{\Gamma \vdash A_1 \vee A_2 \uparrow} [\vee I]$$

Fig. 5. The rules for $\vee$ for intuitionistic natural deduction. In $[\vee L]$, $i \in \{1, 2\}$.

C. *Natural deduction for intuitionistic logic*

If we set $\delta(\lfloor \cdot \rfloor) = +2$ and $\delta(\lceil \cdot \rceil) = -1$, then the synthetic rules derived in **F** for the rule expressions in Fig. 3 describe natural deduction proofs in intuitionistic logic. To prove this claim, we take the rules in Fig. 4 as the formal definition of natural deduction [7].

Let $\Gamma \cup \{C\}$ be a set of propositional formulas and assume that all $\delta(\lfloor \cdot \rfloor) = +2$ and $\delta(\lceil \cdot \rceil) = -1$. The two judgments in Fig. 4 will be encoded as follows. The up-arrow judgment $\Gamma \vdash C \uparrow$ is encoded using $\vdash \lfloor \Gamma \rfloor, \lceil C \rceil$. The down-arrow judgment $\Gamma \vdash C \downarrow$ is encode using $\vdash \lfloor \Gamma \rfloor, \overline{\lceil C \rceil}$.

Consider, for example, the following derivation using the $(\supset L^m)$ rule in Fig. 3.

$$\dfrac{\dfrac{\dfrac{\dfrac{\vdash \overline{\lfloor A \supset B \rfloor}, \Upsilon}{\Downarrow \lfloor A \supset B \rfloor \vdash \Upsilon}\; 2 \quad \dfrac{\vdash \lceil A \rceil, \Upsilon}{\vdash \lceil A \rceil \Downarrow \Upsilon}\; 3}{\Downarrow \lfloor A \supset B \rfloor \mapsto \lceil A \rceil \vdash \Upsilon} \qquad \dfrac{}{\vdash \lfloor B \rfloor \Downarrow \overline{\lfloor B \rfloor}, \Upsilon}\; 4}{\Downarrow (\lfloor A \supset B \rfloor \mapsto \lceil A \rceil) \Mapsto \lfloor B \rfloor \vdash \overline{\lfloor B \rfloor}, \Upsilon}}{\vdash \overline{\lfloor B \rfloor}, \Upsilon}\; 1$$

This derivation uses the **F** rules (1) *decide*, (2) *debit*$_2$, (3) *release*, and (4) *initR*. The associated synthetic inference rule is thus

$$\dfrac{\vdash \overline{\lfloor A \supset B \rfloor}, \Upsilon \quad \vdash \lceil A \rceil, \Upsilon}{\vdash \overline{\lfloor B \rfloor}, \Upsilon}$$

In this example, since Υ can only contain atomic expressions of the form $\lfloor \cdot \rfloor$, we can write $\lfloor \Gamma \rfloor$ for Υ. Thus, we have correctly captured the $[\supset E]$ inference rule in Fig. 4.

Deciding on (Id_1) and (Id_2), respectively, yields

$$\dfrac{\dfrac{\dfrac{\vdash \overline{\lfloor B \rfloor}, \Upsilon}{\Downarrow \lfloor B \rfloor \vdash \Upsilon}\; \textit{debit}_2 \quad \dfrac{}{\Downarrow \lceil B \rceil \vdash \lceil B \rceil, \Upsilon}\; \textit{initL}}{\Downarrow \lfloor B \rfloor \times \lceil B \rceil \vdash \lceil B \rceil, \Upsilon}}{\vdash \lceil B \rceil, \Upsilon}$$

$$\dfrac{\dfrac{\dfrac{\dfrac{}{\Downarrow \mathbf{1} \vdash \Upsilon} \quad \dfrac{\vdash \lceil B \rceil, \Upsilon}{\vdash \lceil B \rceil \Downarrow \Upsilon}\; \textit{release}}{\Downarrow \mathbf{1} \mapsto \lceil B \rceil \vdash \Upsilon} \qquad \dfrac{}{\vdash \lfloor B \rfloor \Downarrow \overline{\lfloor B \rfloor}, \Upsilon}\; \textit{initL}}{\Downarrow \mathbf{1} \mapsto \lceil B \rceil \Mapsto \lfloor B \rfloor \vdash \overline{\lfloor B \rfloor}, \Upsilon}}{\vdash \overline{\lfloor B \rfloor}, \Upsilon}$$

and these yield the two synthetic rules

$$\dfrac{\vdash \overline{\lfloor B \rfloor}, \Upsilon}{\vdash \lceil B \rceil, \Upsilon} \quad \text{and} \quad \dfrac{\vdash \lceil B \rceil, \Upsilon}{\vdash \overline{\lfloor B \rfloor}, \Upsilon}\,.$$

These rules encode the natural deduction rules $[M]$ and $[S]$ rules, respectively.

Consider the synthetic rules using the $(\vee L^a)$ rule in Fig. 3 for a final example.

$$\dfrac{\dfrac{\vdash \overline{\lfloor A \vee B \rfloor}, \Upsilon}{\Downarrow \lfloor A \vee B \rfloor \vdash \Upsilon}\; \textit{debit}_2 \quad \dfrac{\dfrac{\vdash \lfloor A \rfloor, \Delta, \Upsilon \quad \vdash \lfloor B \rfloor, \Delta, \Upsilon}{\vdash \lfloor A \rfloor + \lfloor B \rfloor, \Delta, \Upsilon}}{\vdash \lfloor A \rfloor + \lfloor B \rfloor \Downarrow \Delta, \Upsilon}\; \textit{release}}{\Downarrow \lfloor A \vee B \rfloor \mapsto \lfloor A \rfloor + \lfloor B \rfloor \vdash \Delta, \Upsilon}$$

Note that Δ could be either $\overline{\lfloor C \rfloor}$ or $\lceil C \rceil$ for some formula C. As a result, the left introduction for disjunction can appear in either the $\downarrow$ or $\uparrow$ style judgments. Thus, this synthetic inference rule faithfully captures the $(\vee E)$ inference rule in Fig. 5.

Let $\Gamma \vdash_{nj} C \uparrow$ and $\Gamma \vdash_{nj} C \downarrow$ denote, respectively, the facts that $\Gamma \vdash C \uparrow$ and $\Gamma \vdash C \downarrow$ are provable using the rules in Fig. 4 and 5. Let $\mathcal{R}_{nj}$ be the rules in Fig. 3. The following proposition holds.

Proposition 9: Let $\Gamma \cup \{C\}$ be a set of object-level formulas and assume that all $\delta(\lceil \cdot \rceil) = -1$ and $\delta(\lfloor \cdot \rfloor) = +2$. Then $\Gamma \vdash_{nj} C \uparrow$ if and only if $\vdash \lfloor \Gamma \rfloor, \lceil C \rceil$ is provable using $\mathcal{R}_{nj}$. and $\Gamma \vdash_{nj} C \downarrow$ if and only if $\vdash \lfloor \Gamma \rfloor, \overline{\lfloor C \rfloor}$ is provable using $\mathcal{R}_{nj}$.

If the disjunction $\vee$ is removed, then derivations are considered *normal* (also, *cut free*) if they do not use switch rule ($[S]$ rule in Fig. 4). Thus, normal proofs can be encoded for such formulas simply by removing (Id_2) from consideration in Proposition 9. See [6] for a similar result but where **F** is replaced by a linear logic proof system.

D. Sequent calculi for classical and intuitionistic logic

When the polarities attributed to $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$ are both negative, the synthetic rules based on the rules in Fig. 3 encode sequent calculus proofs. For an example, if we assign $\delta(\lfloor \cdot \rfloor) = -2$ and $\delta(\lceil \cdot \rceil) = -1$, then the implication left rule $(\supset L^m)$ yields the following synthetic inference rule.

$$\dfrac{\vdash \lceil A \rceil, \lfloor A \supset B \rfloor, \Upsilon \quad \vdash \lfloor B \rfloor, \lfloor A \supset B \rfloor, \Delta, \Upsilon}{\vdash \lfloor A \supset B \rfloor, \Delta, \Upsilon}$$

This synthetic inference rule encodes the sequent calculus rule (assuming that Δ is the multiset consisting of one occurrence of $\lceil C \rceil$).

$$\dfrac{A \supset B, \Gamma \vdash A \quad A \supset B, B, \Gamma \vdash C}{A \supset B, \Gamma \vdash C}$$

If we change the bias assignment so that $\delta(\lfloor \cdot \rfloor) = -2$ and $\delta(\lceil \cdot \rceil) = -2$ and consider the same implication-left inference rule, then the same development holds except that the multiset Δ is empty since all atoms belong to the classical realm: the schema variable Υ will hold atoms of both the form $\lfloor \cdot \rfloor$ and $\lceil \cdot \rceil$. As a result, we get the derived inference rule

$$\dfrac{A \supset B, \Gamma \vdash A, \Psi \quad A \supset B, B, \Gamma \vdash \Psi}{A \supset B, \Gamma \vdash \Psi}\,.$$

As with the natural deduction calculus, the (Id_1) and (Id_2) rules have special roles. In particular, using *decide* with them yields the following.

$$\dfrac{\dfrac{}{\Downarrow \lfloor C \rfloor \vdash \lfloor C \rfloor, \Upsilon}\; \textit{initL} \quad \dfrac{}{\Downarrow \lceil C \rceil \vdash \lceil C \rceil, \Upsilon}\; \textit{initL}}{\Downarrow \lfloor C \rfloor \times \lceil C \rceil \vdash \lfloor C \rfloor, \lceil C \rceil, \Upsilon}\; \textit{decide } Id_1$$

$$\dfrac{\dfrac{}{\Downarrow \mathbf{1} \vdash \Upsilon} \quad \vdash \lceil C \rceil, \Upsilon \quad \vdash \lfloor C \rfloor, \Delta, \Upsilon}{\Downarrow \mathbf{1} \Mapsto \lceil C \rceil \mapsto \lfloor C \rfloor \vdash \Delta, \Upsilon}\; \textit{decide } Id_2$$

These are the following synthetic rules

$$\dfrac{}{\vdash \lfloor C \rfloor, \lceil C \rceil, \Upsilon} \qquad \dfrac{\vdash \lceil C \rceil, \Upsilon \quad \vdash \lfloor C \rfloor, \Delta, \Upsilon}{\vdash \Delta, \Upsilon}$$

In the intuitionistic setting, the variable Υ contains only $\lfloor \cdot \rfloor$ atomic expressions while Δ contains only a single expression and that is of the form $\lceil \cdot \rceil$. Thus, (Id_1) and (Id_2) encode the *init* and *cut* rules of sequent calculus. (This encoding works for both intuitionistic and classical logic.)

Given this discussion, the following has a direct proof.

Proposition 10 (Negative bias encodes sequent calculus): If $\delta(\lfloor \cdot \rfloor) = -2$ and $\delta(\lceil \cdot \rceil) = -1$ then the rules in Fig. 3 encode a sequent calculus proof system (similar to Gentzen's **LJ** proof system) which is complete for intuitionistic logic. If, however,

$$
\begin{array}{lrl}
 & \lfloor A \supset B \rfloor \times \lceil C \rceil \mapsto & \lceil A \rceil \mapsto \lfloor B \rfloor \times \lceil C \rceil \\
 & \lceil A \supset B \rceil \mapsto & \lfloor A \rfloor \times \lceil B \rceil \\
 & \lfloor A \wedge B \rfloor \mapsto & \lfloor A \rfloor \\
 & \lceil A \wedge B \rceil \mapsto & \lceil A \rceil + \lceil B \rceil \\
 & \lfloor A \wedge B \rfloor \mapsto & \lfloor B \rfloor \\
 & \lfloor A \vee B \rfloor \mapsto & \lfloor A \rfloor + \lfloor B \rfloor \\
 & \lceil A \vee B \rceil \mapsto & \lceil A \rceil \\
 & \lceil A \vee B \rceil \mapsto & \lceil B \rceil \\
 & \lfloor \bot \rfloor \mapsto & \mathbf{0} \\
 & \lceil \top \rceil \mapsto & \mathbf{0} \\
(Id_1) & \lfloor C \rfloor \times \lceil C \rceil & \\
(Id_2) & \lceil A \rceil \mapsto \lceil C \rceil \mapsto & \lfloor C \rfloor \times \lceil A \rceil \\
(LW) & \lfloor B \rfloor \mapsto & \mathbf{1} \\
(LC) & \lfloor B \rfloor \mapsto & \lfloor B \rfloor \times \lfloor B \rfloor
\end{array}
$$

Fig. 6. Some rewrite rules used to specify sequent calculus proofs in intuitionistic logic entirely in the linear realm.

we change the bias assignment so that $\delta(\lceil \cdot \rceil) = -2$, then the rules in Fig. 3 encode a sequent calculus proof system (similar to Gentzen's **LK** proof system) which is complete for classical logic.

By using Propositions 5, 9, and 10, we can conclude immediately that if a formula has a natural deduction proof then it has a sequent calculus proof, since the only difference between these two encodings is the use of the debit rules.

It is worth noting that if $\delta(\cdot)$ is modified so that for some atomic expressions A, the value of $\delta(A)$ changes from -1 to -2, then proofs in **B** remain proofs. Thus, it is immediate that sequent provability in intuitionistic logic yields sequent provability in classical logic.

It is possible to encode the sequent calculus for both classical and intuitionistic logic at a more primitive level: that is, by using only the linear realm. In particular, consider the specification in Fig. 6. If $\delta(\lfloor \cdot \rfloor) = -1$ and $\delta(\lceil \cdot \rceil) = -1$, then these rules yield sequent calculus proofs for intuitionistic logic. Dropping the use of the classical realm affected this specification in two ways. First, we needed to add explicit weakening and contraction rules for left formula (the rules (LW) and (LC), respectively). Second, in encoding the implication-left rule and the cut rule, the occurrences of the right-side formula must be explicitly addressed in the rule's specification. In order to capture classical sequent calculus, we can modify the rules in Fig. 6 by replacing the left rule for implication and the (Id_2) rule with the rules

$$
\begin{array}{c}
\lfloor A \supset B \rfloor \mapsto \lceil A \rceil \mapsto \lfloor B \rfloor \\
\mathbf{1} \mapsto \lceil C \rceil \mapsto \lfloor C \rfloor
\end{array}
$$

and by adding the following explicit rules for weakening and contraction for right tagged formulas.

$$
(RW)\ \lceil B \rceil \mapsto \mathbf{1} \qquad (RC)\ \lceil B \rceil \mapsto \lceil B \rceil \times \lceil B \rceil
$$

E. Alternative encodings of proof rules

Fig. 7 contains alternative specifications of the introduction rules for some propositional logic constants. In particular,

$$
\begin{array}{lrl}
(\supset L^a) & \lfloor A \supset B \rfloor \mapsto & \lceil A \rceil \\
(\supset R^a) & \lceil A \supset B \rceil \mapsto & \lfloor A \rfloor \\
(\supset R^a) & \lceil A \supset B \rceil \mapsto & \lceil B \rceil \\
(\wedge L^m) & \lfloor A \wedge B \rfloor \mapsto & \lfloor A \rfloor \times \lfloor B \rfloor \\
(\wedge R^m) & \lceil A \wedge B \rceil \mapsto & \lceil A \rceil \mapsto \lceil B \rceil \\
(\vee L^m) & \lfloor A \vee B \rfloor \mapsto & \lfloor A \rfloor \mapsto \lfloor B \rfloor \\
(\vee R^m) & \lceil A \vee B \rceil \mapsto & \lceil A \rceil \times \lceil B \rceil \\
(\bot L^m) & \lfloor \bot \rfloor \mapsto & \mathbf{1} \\
(\top R^m) & \lceil \top \rceil \mapsto & \mathbf{1}
\end{array}
$$

Fig. 7. Some alternative version of inference rules.

$$
\begin{array}{l}
\mathbf{1} \mapsto \lfloor A \wedge B \rfloor \mapsto \lceil A \rceil \mapsto \lceil B \rceil \\
\mathbf{1} \mapsto \lceil A \wedge B \rceil \mapsto \lfloor A \rfloor \\
\mathbf{1} \mapsto \lceil A \wedge B \rceil \mapsto \lfloor B \rfloor \\
\mathbf{1} \mapsto \lfloor A \vee B \rfloor \mapsto \lceil A \rceil \\
\mathbf{1} \mapsto \lfloor A \vee B \rfloor \mapsto \lceil B \rceil \\
\mathbf{1} \mapsto \lceil A \vee B \rceil \mapsto \lfloor A \rfloor \mapsto \lfloor B \rfloor \\
\mathbf{1} \mapsto \lfloor A \supset B \rfloor \mapsto \lfloor A \rfloor \\
\mathbf{1} \mapsto \lfloor A \supset B \rfloor \mapsto \lceil B \rceil \\
\mathbf{1} \mapsto \lceil A \supset B \rceil \mapsto \lceil A \rceil \mapsto \lfloor B \rfloor
\end{array}
$$

Fig. 8. Specification of the free deduction for classical logic.

while Fig. 3 provides multiplicative rules for implication and additive rules for conjunction, disjunction, true, and false, in Fig. 7, we find additive rules for implication and multiplicative rules for conjunction, disjunction, true, and false. As is well known, the presence of the structural rules (of weakening and contraction) allows some pairing of these rules to be inter-admissible.

If we switch from the additive rules for conjunction ($\wedge L^a$ and $\wedge R^a$ in Fig. 3) to the multiplicative rules ($\wedge L^m$ and $\wedge R^m$ in Fig. 7), then the conjunction elimination rule of intuitionistic natural deduction can be computed as follows.

$$
\dfrac{\dfrac{\vdash \overline{\lfloor A \wedge B \rfloor}, \Upsilon}{\Downarrow \lfloor A \wedge B \rfloor \vdash \Upsilon}\ \mathit{debit}_2 \quad \dfrac{\dfrac{\vdash \lfloor A \rfloor, \lfloor B \rfloor, \Delta, \Upsilon}{\vdash \lfloor A \rfloor \times \lfloor B \rfloor, \Delta, \Upsilon}}{\vdash \lfloor A \rfloor \times \lfloor B \rfloor \Downarrow \Delta, \Upsilon}\ \mathit{release}}{\Downarrow \lfloor A \wedge B \rfloor \mapsto \lfloor A \rfloor \times \lfloor B \rfloor \vdash \Delta, \Upsilon}
$$

Since Δ could be either $\overline{\lceil C \rceil}$ or $\lceil C \rceil$ for some formula C, the left-introduction for conjunction can appear in either the $\downarrow$ or $\uparrow$ style judgments.

$$
\frac{\Gamma \vdash A \wedge B \downarrow \quad \Gamma, A, B \vdash C \uparrow\!(\downarrow)}{\Gamma \vdash C \uparrow\!(\downarrow)}
$$

This natural deduction rule is an example of a *generalized elimination rule* [8], [9].

F. Free deduction in classical logic

Given the interpretation of (Id_1) and (Id_2) as the $[M]$ and $[S]$ inference rules in natural deduction, it is tempting to consider both $\lfloor B \rfloor$ and $\overline{\lceil B \rceil}$ and both $\lceil B \rceil$ and $\overline{\lfloor B \rfloor}$ as equivalent in some sense. Such possible equivalences do not immediately apply to rules, however, since rule expressions do

$$
\begin{array}{l}
\lceil A \wedge B \rceil \times \lfloor A \rfloor \times \lfloor B \rfloor \\
\lfloor A \wedge B \rfloor \times \lceil A \rceil \\
\lfloor A \wedge B \rfloor \times \lceil B \rceil \\
\lceil A \vee B \rceil \times \lfloor A \rfloor \\
\lceil A \vee B \rceil \times \lfloor B \rfloor \\
\lfloor A \vee B \rfloor \times \lceil A \rceil \times \lceil B \rceil \\
\lceil A \supset B \rceil \times \lceil A \rceil \\
\lceil A \supset B \rceil \times \lfloor B \rfloor \\
\lfloor A \supset B \rfloor \times \lfloor A \rfloor \times \lceil B \rceil
\end{array}
$$

Fig. 9. Several simple expressions provable from $\mathcal{C}$.

not contain debt expressions. It might be possible, however, to link proofs using a rule of the form $\lfloor A \rfloor \mapsto \lfloor B \rfloor$ with a proof using a rule of the form $\mathbf{1} \mapsto \lceil A \rceil \mapsto \lfloor B \rfloor$. We illustrate such considerations in this section.

The Free Deduction proof system [10] can be encoded as follows. Let $\mathcal{FD}$ be the set of rules that results from taking the rules in Fig. 8 along with (Id_1) the following variant of (Id_2):

$$(Id_3) \qquad \mathbf{1} \mapsto \lceil C \rceil \mapsto \lfloor C \rfloor.$$

Also, let $\mathcal{C}$ be composed of the rules in Fig. 3. When using both of these sets of rules, we assume that $\delta(\lfloor\cdot\rfloor) = \delta(\lceil\cdot\rceil) = -2$. As we have seen, under this bias assignment, the rules in $\mathcal{C}$ encode a classical sequent system. It is a simple exercise to show that all of the expressions in Fig. 9 are provable from $\mathcal{C}$. Also, note the strong similarities between the rules in Fig. 8 and the expressions in Fig. 9: by dropping the $\mathbf{1} \mapsto$ prefix, changing the remaining occurrences of $\mapsto$ to $\times$, and flipping the left and right tags, we can convert rules in Fig. 8 to expressions in Fig. 9.

It is easy to show that every use of a rule in $\mathcal{FD}$ can be emulated by deciding on an expression in Fig. 9. For example, the synthetic rule that results from the first rule in Fig. 8 is

$$\frac{\vdash \lfloor A \wedge B \rfloor, \Delta_1 \quad \vdash \lceil A \rceil, \Delta_2 \quad \vdash \lceil B \rceil, \Delta_3}{\vdash \Delta_1, \Delta_2, \Delta_3}$$

This inference rule can be modeled in **B** by deciding on the first expression in Fig. 9 and using (Id_1) three times (and with shifting the polarity to $\delta(\lfloor\cdot\rfloor) = \delta(\lceil\cdot\rceil) = +2$):

$$\cfrac{\cfrac{\vdash \overline{\lceil A \wedge B \rceil}, \Delta_1 \quad \vdash \overline{\lceil A \rceil}, \Delta_2 \quad \vdash \overline{\lceil B \rceil}, \Delta_3}{\lceil A \wedge B \rceil \times \lfloor A \rfloor \times \lfloor B \rfloor \vdash \Delta_1, \Delta_2, \Delta_3}}{\vdash \Delta_1, \Delta_2, \Delta_3}$$

By using decide on the rule (Id_3) on all three premises above, we can build a **B** derivation that flips the debt $\overline{\lceil A \wedge B \rceil}$ to the atomic expression $\lfloor A \wedge B \rfloor$ (as in the $[S]$ inference rule in Section V-C). It is now a simple matter to use the clip-elimination theorem to remove the intermediate lemmas listed in Fig. 9 for direct **B**-proofs. Once we have such **B**-proofs, Theorem 8 can provide an **F** proof corresponding to classical sequent calculus proof.

$$
\begin{array}{rl}
(\multimap L) & \lfloor A \multimap B \rfloor \mapsto \lceil A \rceil \mapsto \lfloor B \rfloor. \\
(\multimap R) & \lceil A \multimap B \rceil \mapsto \lfloor A \rfloor \times \lceil B \rceil. \\
(\otimes L) & \lfloor A \otimes B \rfloor \mapsto \lfloor A \rfloor \times \lfloor B \rfloor. \\
(\otimes R) & \lceil A \otimes B \rceil \mapsto \lceil A \rceil \mapsto \lceil B \rceil. \\
(\& L_1) & \lfloor A \,\&\, B \rfloor \mapsto \lfloor A \rfloor. \\
(\& R) & \lceil A \,\&\, B \rceil \mapsto \lceil A \rceil + \lceil B \rceil. \\
(\& L_2) & \lfloor A \,\&\, B \rfloor \mapsto \lfloor B \rfloor. \\
(\oplus R_1) & \lceil A \oplus B \rceil \mapsto \lceil A \rceil. \\
(\oplus L) & \lfloor A \oplus B \rfloor \mapsto \lfloor A \rfloor + \lfloor B \rfloor. \\
(\oplus R_2) & \lceil A \oplus B \rceil \mapsto \lceil B \rceil. \\
(\invamp L) & \lfloor A \invamp B \rfloor \mapsto \lfloor A \rfloor \mapsto \lfloor B \rfloor. \\
(\invamp R) & \lceil A \invamp B \rceil \mapsto \lceil A \rceil \times \lceil B \rceil. \\
(1L) & \lfloor 1 \rfloor. \\
(1R) & \lceil 1 \rceil \Mapsto \mathbf{1}. \\
(\bot\, L) & \lfloor \bot \rfloor \Mapsto \mathbf{1}. \\
(\bot\, R) & \lceil \bot \rceil. \\
(0L) & \lfloor 0 \rfloor \mapsto \mathbf{0}. \\
(\top R) & \lceil \top \rceil \mapsto \mathbf{0}. \\
(!\, L) & \lfloor !\, B \rfloor \mapsto \lfloor\lfloor B \rfloor\rfloor. \\
(!\, R) & \lceil !\, B \rceil \mapsto \lceil B \rceil \Mapsto \mathbf{1}. \\
(?L) & \lfloor ?B \rfloor \mapsto \lfloor B \rfloor \Mapsto \mathbf{1}. \\
(?R) & \lceil ?B \rceil \mapsto \lceil\lceil B \rceil\rceil. \\
(\mathit{derL}) & \lfloor\lfloor B \rfloor\rfloor \mapsto \lfloor B \rfloor. \\
(\mathit{derR}) & \lceil\lceil B \rceil\rceil \mapsto \lceil B \rceil.
\end{array}
$$

Fig. 10. Specification of linear logic.

G. Linear logic

Fig. 10 contains a specification for linear logic. This specification makes use of *four* tags: $\lfloor\cdot\rfloor$, $\lceil\cdot\rceil$, $\lfloor\lfloor\cdot\rfloor\rfloor$, and $\lceil\lceil\cdot\rceil\rceil$. Here, $\lfloor\cdot\rfloor$ and $\lceil\cdot\rceil$ construct atomic expressions that should be in the linear realm while $\lfloor\lfloor\cdot\rfloor\rfloor$ and $\lceil\lceil\cdot\rceil\rceil$ construct atomic expressions that should be in the classical realm. A sequent calculus for linear logic arises when we use the bias assignment $\delta(\lfloor\cdot\rfloor) = \delta(\lceil\cdot\rceil) = -1$ and $\delta(\lfloor\lfloor\cdot\rfloor\rfloor) = \delta(\lceil\lceil\cdot\rceil\rceil) = -2$.

H. Quantification

Some of the earliest work on logic frameworks (for example, using λProlog [11] and the dependently typed LF [12], [13]) provided elegant approaches to the treatment of quantifiers in the specification of proof systems. The essence of these treatments of quantifiers is described via the notion of *binder mobility* [14], a concept we illustrate briefly here. We first extended the grammar of E and R formulas to allow both $\boldsymbol{Q}\, x.(E\ x)$ and $\boldsymbol{Q}\, x.(R\ x)$, where $\boldsymbol{Q}\, x.$ is a binder for x over expressions and rules. Next, we need to add to sequents a place for expression-level binders to move. To this end, we attach a variable-binding context Σ to all sequents. Thus, sequents have the structure $\Sigma : \Gamma \vdash \Delta$ and $\Sigma : \Downarrow\Gamma \vdash \Delta$. In both of these cases, Σ is a list of distinct variables, all with scope intended over the formulas in the respective sequent. We assume the usual notions of α, β, and η conversion. The following two rules can be added to **B** to treat quantifiers.

$$\frac{\Sigma, x : \Gamma \vdash E\ x, \Delta}{\Sigma : \Gamma \vdash \boldsymbol{Q}\, x.E\ x, \Delta} \qquad \frac{\Sigma : \Gamma, R\ t \vdash \Delta \quad t \text{ is a } \Sigma\text{-term}}{\Sigma : \Gamma, \boldsymbol{Q}\, x.R\ x \vdash \Delta}$$

In the first rule, we assume that x is not already bound by Σ. In that rule, the expression-level binder for x in the conclusion is moved to a sequent-level binder for x in the premise. The proviso in the second inference rule means that the free variables of the (first-order) term t are all taken from Σ.

Finally, to illustrate how quantifiers can be used to specify rules, we first explicitly quantify over schema variables in the specification of rules. For example, the rule $(\supset L^m)$ in Fig. 3 should be written more explicitly as

$$\boldsymbol{Q}\, A.\boldsymbol{Q}\, B.\lfloor A \supset B \rfloor \mapsto \lceil A \rceil \Rrightarrow \lfloor B \rfloor$$

Adding universal and existential quantification to the intuitionistic and classical logic of Section V-B can be done using the (closed) R-expressions

$$\begin{array}{rlcl} \boldsymbol{Q}\, B.\boldsymbol{Q}\, t. & \lfloor \forall x.Bx \rfloor & \mapsto & \lfloor Bt \rfloor \\ \boldsymbol{Q}\, B. & \lceil \forall x.Bx \rceil & \mapsto & \boldsymbol{Q}\, x.\lceil Bx \rceil \\ \boldsymbol{Q}\, B. & \lfloor \exists x.Bx \rfloor & \mapsto & \boldsymbol{Q}\, x.\lfloor Bx \rfloor \\ \boldsymbol{Q}\, B.\boldsymbol{Q}\, t. & \lceil \exists x.Bx \rceil & \mapsto & \lceil Bt \rceil \end{array}$$

VI. Related work

The two-phase proof system **F** resembles *uniform proofs*, which were used to describe logic programming as the search for proofs in a two-phase proof system that alternated between a *goal-reduction* phase and a *backchaining* phase [15]. Andreoli's focused proof system [16] for Girard's linear logic [2] also inspired design aspects of **PSF**. The closest related work, however, is the following collection of papers that have used linear logic as a logical framework for specifying proof systems. The author showed how a version of linear logic based on only the negative connectives could be used to specify Gentzen-style sequent calculus and natural deduction proof systems [17]. Nigam, Pimentel, and others significantly extended such specifications, especially once subexponentials were added to linear logic [18], [19], [20], [6], [21], [22], [23]. Implementations and formal results surrounding such linear logic specifications have also been built [24], [25]. A design goal for **PSF** was to use it to replace linear logic as the meta-logic while attempting to find the fewest features of linear logic that made it successful for specifying proof systems.

VII. Conclusion

The state of the search for proofs in classical and intuitionistic logic can be viewed as a collection of sheets of paper, each containing assumptions and a conclusion: such sheets denote a gap in the proof to be completed. An inference rule is encoded in reverse as a rule for rewriting a sheet into 0 or more other sheets. **PSF** starts with this simple perspective of inference and formalizes inference as the rewriting of collections of multisets of tagged formulas. In doing so, the multiplicative and additive structures behind logical inference are treated as primitive. This framework also uses a bias assignment for tagged formulas that captures the notions of linear and classical realm and of debt. We have also illustrated how **PSF** specifications of inference rules can be used to represent a range of known proof systems modularly. We demonstrated this modularity by showing that one set of rewrite rules can account for sequent calculus and natural deduction proofs in classical and intuitionistic logic.

References

[1] G. Gentzen, "Investigations into logical deduction," in *The Collected Papers of Gerhard Gentzen*, M. E. Szabo, Ed. North-Holland, 1935, pp. 68–131.

[2] J.-Y. Girard, "Linear logic," *Theoretical Computer Science*, vol. 50, no. 1, pp. 1–102, 1987.

[3] K. Chaudhuri, F. Pfenning, and G. Price, "A logical characterization of forward and backward chaining in the inverse method," *J. of Automated Reasoning*, vol. 40, no. 2-3, pp. 133–177, 2008.

[4] D. Miller and J.-H. Wu, "A positive perspective on term representations," in *31st EACSL Annual Conference on Computer Science Logic (CSL 2023)*, B. Klin and E. Pimentel, Eds., LIPIcs 252, 2023, pp. 3:1–3:21.

[5] C. Liang and D. Miller, "A focused approach to combining logics," *Annals of Pure and Applied Logic*, vol. 162, no. 9, pp. 679–697, 2011.

[6] V. Nigam and D. Miller, "A framework for proof systems," *J. of Automated Reasoning*, vol. 45, no. 2, pp. 157–188, 2010. [Online]. Available: http://springerlink.com/content/m12014474287n423/

[7] W. Sieg and J. Byrnes, "Normal natural deduction proofs (in classical logic)," *Studia Logica*, vol. 60, no. 1, pp. 67–106, 1998.

[8] P. Schroeder-Heister, "A natural extension of natural deduction," *Journal of Symbolic Logic*, vol. 49, no. 4, pp. 1284–1300, 1984.

[9] J. von Plato, "Natural deduction with general elimination rules," *Archive for Mathematical Logic*, vol. 40, no. 7, pp. 541–567, 2001.

[10] M. Parigot, "Free deduction: An analysis of "computations" in classical logic," in *Proceedings of the First Russian Conference on Logic Programming*, ser. LNAI, no. 592. Springer, 1992, pp. 361–380.

[11] A. Felty and D. Miller, "Specifying theorem provers in a higher-order logic programming language," in *Ninth International Conference on Automated Deduction*, E. Lusk and R. Overbeck, Eds., LNCS 310. Argonne, IL: Springer, May 1988, pp. 61–80.

[12] R. Harper, F. Honsell, and G. Plotkin, "A framework for defining logics," *Journal of the ACM*, vol. 40, no. 1, pp. 143–184, 1993.

[13] A. Felty and D. Miller, "Encoding a dependent-type λ-calculus in a logic programming language," in *Proc. of the 1990 Conference on Automated Deduction*, M. Stickel, Ed., LNAI 449, Springer, 1990, pp. 221–235.

[14] D. Miller, "Mechanized metatheory revisited," *Journal of Automated Reasoning*, vol. 63, no. 3, pp. 625–665, Oct. 2019.

[15] D. Miller, G. Nadathur, F. Pfenning, and A. Scedrov, "Uniform proofs as a foundation for logic programming," *Annals of Pure and Applied Logic*, vol. 51, no. 1–2, pp. 125–157, 1991.

[16] J.-M. Andreoli, "Logic programming with focusing proofs in linear logic," *J. of Logic and Computation*, vol. 2, no. 3, pp. 297–347, 1992.

[17] D. Miller, "Forum: A multiple-conclusion specification logic," *Theoretical Computer Science*, vol. 165, no. 1, pp. 201–232, 1996.

[18] E. G. Pimentel, "Lógica linear e a especificação de sistemas computacionais," Ph.D. dissertation, Universidade Federal de Minas Gerais, Belo Horizonte, M.G., Brasil, Dec. 2001, written in English.

[19] V. Nigam, "Exploiting non-canonicity in the sequent calculus," Ph.D. dissertation, Ecole Polytechnique, Sep. 2009. [Online]. Available: http://www.nigam.info/thesis/Vivek_Nigam_phd.pdf

[20] V. Nigam and D. Miller, "Algorithmic specifications in linear logic with subexponentials," in *ACM SIGPLAN Conference on Principles and Practice of Declarative Programming (PPDP)*, A. Porto and F. J. López-Fraguas, Eds. ACM, 2009, pp. 129–140.

[21] V. Nigam, E. Pimentel, and G. Reis, "Specifying proof systems in linear logic with subexponentials," in *Proceedings of the Fifth Logical and Semantic Frameworks, with Applications Workshop (LSFA 2010)*, ser. ENTCS, vol. 269. Elsevier, 2011, pp. 109–123.

[22] D. Miller and E. Pimentel, "A formal framework for specifying sequent calculus proof systems," *Theoretical Computer Science*, vol. 474, pp. 98–116, 2013. [Online]. Available: http://hal.inria.fr/hal-00787586

[23] V. Nigam, E. Pimentel, and G. Reis, "An extended framework for specifying and reasoning about proof systems," *J. of Logic and Computation*, 2014.

[24] G. Reis and V. Nigam, "TATU: Online system for reasoning about sequent calculus specifications in linear logic with subexponentials," http://www.logic.at/people/giselle/tatu/, Mar. 2010.

[25] C. Olarte, E. Pimentel, and C. Rocha, "A rewriting logic approach to specification, proof-search, and meta-proofs in sequent systems," *Journal of Logical and Algebraic Methods in Programming*, vol. 130, 2023.